Semantic Integrator: Semi-Automatically Enhancing Social Semantic Web Environments

Steffen Lohmann, Philipp Heim, Jürgen Ziegler

University of Duisburg-Essen
Dep. of Informatics and Applied Cognitive Science
Lotharstrasse 65, 47057 Duisburg, Germany
{heim, lohmann, ziegler}@interactivesystems.info

Abstract: Large amounts of information from various sources have often to be considered when collaboratively developing semantic structures. Examining all relevant information can be very demanding and time consuming. Thus, methods and tools are needed that assist in the integration of this heterogeneous and distributed information. Based on an approach that uses Social Software and Semantic Web technology in requirements engineering, this paper describes the general concept and architecture of the Semantic Integrator, a tool that aims at visually support the integration of distributed information into semantic structures.

1 Introduction

The comprehensive collection of requirements is essential to successful software development. However, considering all sources of requirements and collect, analyze and merge the gathered information is challenging, particularly if the user groups are very large and spatially distributed. Semantic Web and Web 2.0 technologies open up new opportunities to better cope with these difficulties. Within the SoftWiki research project [Sof07], a web based collaborative environment is developed that fosters stakeholder participation in early requirements engineering. The SoftWiki philosophy follows the notion of the Social Semantic Web: Participation should be as easy as possible and semantically structured at the same time.

Though this "Wiki Way" [LC01] of requirements elicitation lowers the participation barrier and increases stakeholder involvement, large parts of stakeholders may still not have the skills, time, or motivation to actively use the collaborative environment. Furthermore, relevant information may already exist in some form or other but needs to be integrated. Examples are end user statements made in e-mails or webforms, on blogs or discussion boards, as well as existing documents and system descriptions. Thus, we search for ways to enhance Social Semantic Web Environments[1] by integrating these distributed information in an efficient way.

[1] By *Social Semantic Web Environments* we mean community platforms that combine Social Software and Semantic Web technologies (e.g. Semantic Wikis).

In the following we describe the general concept and architecture of the Semantic Integrator, a tool we are currently developing within the SoftWiki project. It aims at visually support the integration of information from diverse sources into an existing semantic structure (e.g. an ontology).

2 Semantic Integration

Three basic principles are at the heart of our approach: (1) The semantic integration should follow a semi-automatic process – manual and automatic activities shall complement each other. (2) The automatic integration should evolutionary improve by learning from the manual integration. (3) The integration process must always remain in the control of the user.

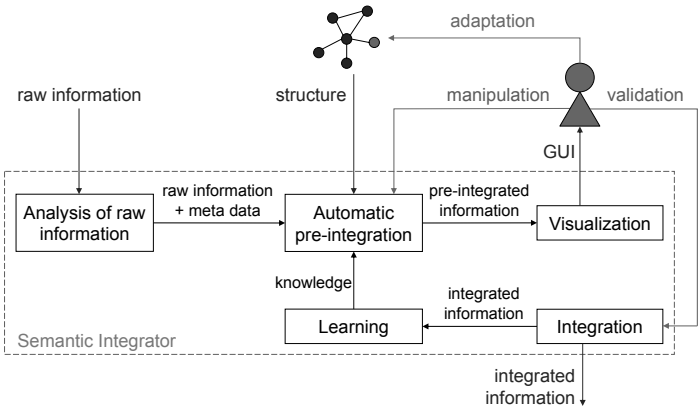

Figure 1: Raw information is analyzed and automatically pre-integrated in the existing structure. The result is then visualized to the user, who interactively manipulates and validates the pre-integration or adapts the underlying structure. Whenever information is manipulated or integrated this leads to a learning step for the next automatic pre-integration.

Usually, two kinds of input sources are provided to the Semantic Integrator (cp. Figure 1): an already existing semantic structure and one or more text documents containing the information that is intended to be integrated into the structure. We subsume the latter under the term *raw information* – though this information may be delivered in a structured way, it usually does not directly fit with the semantic scheme of the existing structure[2]. The Semantic Integrator aims to be able to process various XML-based input formats: The semantic structure may be provided in RDF or OWL, the raw information in XHTML or OpenDocument format. As output, RDF is generated that contains the adapted structure and the information integrated into it. With these standardized XML-based input and out-

[2]Furthermore, the Semantic Integrator may be used as a visual tool that assists in building an initial structure out of the raw information in cases where a semantic structure does not already preexist.

put interfaces, it will be possible to seamlessly plug the Semantic Integrator into Social Semantic Web Environments. The semantic integration process consists of the following components:

2.1 Analysis of Raw Information

Having selected the sources that should be considered for semantic integration, the included raw information is first analyzed. For this purpose, we use several text mining algorithms that work in conjunction with a large reference corpus [HQW02] and that have already been successfully applied in former research projects (see e.g. [ZJB05]).

First, the text is segmented into its single sentences and words, the stop words are eliminated and an index is generated. Typical word usage is derived by comparison with the reference corpus. Additionally, collocations are calculated and compared with the reference corpus. A collocation is the significant co-occurrence of two or more words within a well-defined unit of information (cp. [MS99]). The significant key words that are extracted out of the raw information in this process are then passed to the automatic pre-integration component.

2.2 Automatic Pre-Integration

To reduce the effort to integrate the raw information in the given structure as well as to extract or expand a structure out of the information, the system executes an automatic pre-integration step. In this step, the extracted key words are classified as far as possible according to the preexisting structure. For significant key words that cannot be assigned to any of the existing classes of the structure, suggestions for new classes are provided that might be valuable extensions to the structure.

To adequately integrate raw information in a certain structure we consider an automatic integration possible only to a certain extent. Hence, both the automatic classification of the key words as well as the extensions of the class structure are merely suggestions for the integration of raw information and need to get confirmed, manipulated or rejected by the user employing the Semantic Integrator GUI.

2.3 Visualization

The Semantic Integrator GUI is divided into three areas (cp. Figure 2): Firstly, a tree view, providing a hierarchical visual presentation of the preexisting structure plus the automatically derived suggestions for its extensions. Secondly, a similarity view, using a map-based visualization to show how the raw information is pre-integrated into this structure. And

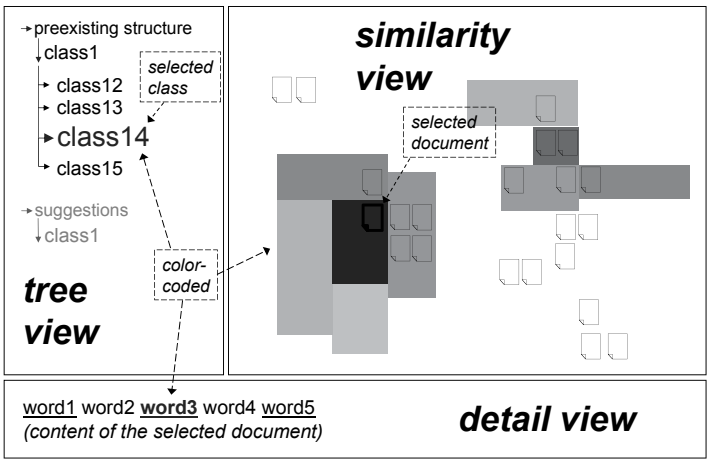

Figure 2: The Semantic Integrator GUI is divided into three areas: a tree view, a similarity view and a detail view.

thirdly, a detail view displaying details for a selected class from the tree view or a selected document from the similarity view.

In order to produce the similarity view we use the Vector Space Model [SAB94]. Hence, every distinct element of the raw information (i.e. every document), is represented by a vector of key words in a vector space spanned by the classes of the preexisting structure and the proposed extensions. The similarity of two documents can now be calculated by taking a similarity measure of the corresponding document vectors using either the scalar product, the cosine similarity measure or the Euclidean distance.

To provide a proper visualization of these similarities we use Self-Organizing Maps (SOMs) [Koh00] to scale our high-dimensional vector space onto a two-dimensional grid. This dimensionality reduction positions similar documents close to each other, which clusters the most related documents and thereby preserves the topology of the input vector space. Such a visualization of the pre-integrated raw information helps the user to identify similarities between documents and to get an overview of related topics.

In addition to the optimal organization of similar documents onto a two-dimensional grid using SOMs, the Semantic Integrator GUI provides color-coded information in accordance to the classification of the key words in the documents. If the user selects a class in the tree view, documents in the similarity view are color-coded depending on whether their key words are assigned to this class or not. If assigned key words are shown in the detail view, they get color-coded, too (cp. figure 2).

2.4 Integration

Equipped with the visualization of the clustered and pre-integrated raw information, the user can then either validate, reject or modify the given pre-integration. The same holds for the automatically generated extension of the class structure. The user can again validate, modify or reject the suggested new classes or build own extensions. We aim to provide intuitive interaction support that enables the user to integrate the raw information by selection, navigation, and drag&drop interaction.

2.5 Learning

The manual integration is then processed in a machine learning step to improve the pre-integration for the next cycle. The objective is to learn classifiers from integration patterns and enhance the quality of the pre-integration. This is implemented by Support Vector Machines (SVM) [Joa98], a supervised learning method that uses an efficient learning algorithm that can represent complex, nonlinear functions. The classification function for every class is learned by training data, in this case the manually integrated raw information. So every manual integration step, such as the handling of a pre-integration or an own classification, serves as training data for the SVM. Based on this data, the SVM calculates the optimal linear separator, a maximum-margin hyperplane, to classify unfamiliar information. Thus, every manual integration evolutionary improves the quality of the automatic pre-integration.

3 Conclusion and Future Work

The Semantic Integrator aims to serve as a semi-automatic tool for organizing, visualizing and integrating distributed information and adapting the underlying structures. Social semantic Web Environments benefit from this approach as information from diverse sources can be considered in the collaborative process in an efficient way. With respect to our requirements engineering approach, we will be able to consider information that is not directly expressed by stakeholders in the collaborative environment.

Future Work includes further development of the Semantic integrator and its incorporation in OntoWiki [ADR06], a tool for collaborative development of ontologies that is used in SoftWiki to gather requirements. Communication between the tools will be realized via REST and SPARQL. This incorporation will enable us to evaluate the achieved semantic integration within a use case in the context of the Social Semantic Web.

References

[ADR06] Sören Auer, Sebastian Dietzold, and Thomas Riechert. OntoWiki - A Tool for Social, Semantic Collaboration. In *Proceedings of the 5th International Semantic Web Conference*, pages 736–749, 2006.

[HQW02] Gerhard Heyer, Uwe Quasthoff, and Christian Wolff. Automatic Analysis of Large Text Corpora - A Contribution to Structuring WEB Communities. In *IICS '02: Proceedings of the Second International Workshop on Innovative Internet Computing Systems*, pages 15–26, London, UK, 2002. Springer-Verlag.

[Joa98] Thorsten Joachims. Text Categorization with Support Vector Machines: Learning with many Relevant Features. In *Proceedings of 10th European Conference on Machine Learning*, pages 137–142, 1998.

[Koh00] Teuvo Kohonen. *Self-Organizing Maps*. Springer, December 2000.

[LC01] Bo Leuf and Ward Cunningham. *The Wiki Way: Quick Collaboration on the Web*. Addison-Wesley, 2001.

[MS99] Christopher D. Manning and Hinrich Schuetze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, Massachusetts, 1999.

[SAB94] Gerard Salton, James Allan, and Chris Buckley. Automatic structuring and retrieval of large text files. *Commun. ACM*, 37(2):97–108, February 1994.

[Sof07] SoftWiki. National research project funded by the German Federal Ministry of Education and Research (BMBF), 2007. http://softwiki.de.

[ZJB05] Jürgen Ziegler, Zoulfa El Jerroudi, and Karsten Böhm. Generating Semantic Contexts from Spoken Conversation in Meetings. In *IUI '05: Proceedings of the 10th International Conference on Intelligent User Interfaces*, pages 290–292, 2005.