Towards Explanation-Aware Social Software: Applying the Mining and Analysis Continuum of Explaining

Martin Atzmueller[1] and Thomas Roth-Berghofer[2,3]

[1] University of Kassel,
Knowledge and Data Engineering Group
Wilhelmshöher Allee 73, 34121 Kassel, Germany

[2] Knowledge Management Department,
German Research Center for Artificial Intelligence (DFKI) GmbH
Trippstadter Straße 122, 67663 Kaiserslautern, Germany

[3] Knowledge-Based Systems Group, Department of Computer Science,
University of Kaiserslautern, P.O. Box 3049, 67653 Kaiserslautern

atzmueller@cs.uni-kassel.de, thomas.roth-berghofer@dfki.de

Abstract. Data mining methods build patterns or models. When presenting these, all or part of the result needs to be explained to the user in order to be understandable and for increasing the user acceptance of the patterns. In doing that, a variety of dimensions in the **M**ining and **A**nalysis **C**ontinuum of **E**xplaining (MACE) needs to be considered, e.g., from concrete to more abstract explanations. This paper discusses the application of the MACE in the context of social software. We consider applications of the proposed approaches in three social software systems, and show how the data mining results can seamlessly be analysed on the presented continuous dimensions and levels.

1 Introduction

Social software systems, e.g., advanced social resource sharing systems such as *Bibsonomy*[4], or popular social networking applications such as *LinkedIn*[5] and *Facebook*[6] are in wide-spread use. Additionally, more and more ubiquitous systems include the user in a more comprehensive way, e.g., [2]. During the utilisation of these systems, a lot of data is acquired that can be applied for improving the user experience of the systems. Interest-driven recommendations, for example, intelligent notifications, or improved ranking of resources are common application areas. For those, data mining methods are usually applied. In order to enable a transparent and trust-enabled approach, explanation capabilities

[4] `www.bibsonomy.org`
[5] `www.linkedin.com`
[6] `www.facebook.com`

provide a viable supplement: Concerning actions of the system, e.g., recommendation, notification, or ranking, often explanations for these actions, relating to the used models, patterns, or parts thereof are requested. If these cannot be fulfilled, acceptance of the actions and the trust in the system is hard to achieve. Appropriate explanation techniques in data mining and analysis are therefore crucial for an effective data mining approach. This is especially relevant for *semantic data mining* and related approaches [4, 19], where background knowledge incorporated by these provides further explanation capabilities, and enables in turn an iterative refinement of the respective explanation knowledge base. Also, appropriate privacy-preserving measures accompanied by suitable explanation techniques can significantly increase the trust in social systems applying data mining techniques. In [5], we present the **M**ining and **A**nalysis **C**ontinuum of **E**xplaining (MACE) providing several explanation dimensions and levels in the context of data mining and analysis. This paper focuses on the application of the MACE in the context of social software, that is, requirements and exemplary questions that can be tackled with corner principles of the MACE. We describe three application scenarios, and discuss the elements of the MACE in this context.

The rest of the paper is structured as follows: Section 2 briefly introduces explanation-aware software design and computing. After that, Section 3 outlines the MACE, including explanation-aware mining and analysis, general explanation goals and kinds, and the other elements of the continuum. Section 4 presents three application scenarios of the new approach in ubiquitous and social environments. Finally, Section 5 concludes the paper with a summary and discusses further interesting options for future research.

2 Explanation-Aware Software Design and Computing

Software systems need the ability to explain reasoning processes and their results as those abilities substantially affect their usability and acceptance. Explanation-aware computing (ExaCt) is the vision of software systems being smart in interactions with their users and Explanation-aware Software Design (EASD) aims at making software systems smarter in this regard. EASD looks at ways to guide software designers and engineers to a purposeful explanation-aware software system by making their designers and engineers explanation-aware. The long-term goal is to provide the respective methods and tools for engineering and improving such explanation capabilities. Here we focus on bringing explanation-awareness techniques to data mining.

The term explanation has been widely investigated in different disciplines such as cognitive science, artificial intelligence, linguistics, philosophy of science, and teaching. All these disciplines consider certain aspects of the term and make clear that there is not only one such concept but a variety of concepts. Explanations are in some sense always answers to questions, may the questions be raised explicitly or not. Explanations are an important vehicle to convey information to understand one another in everyday conversations. They support humans in their decision-making [14].

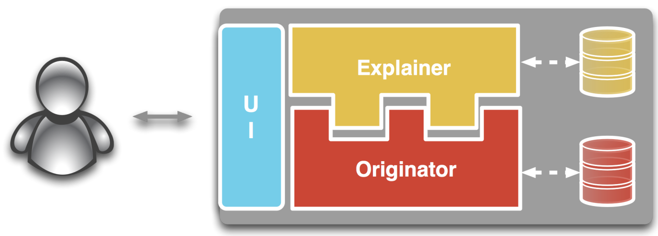

Fig. 1. Communication participants in general explanation scenario [13]

In a general explanation scenario (Figure 1) we distinguish three main participants [13]: the *user* who is corresponding with the software system via its user interface (UI), the *originator*, i.e., the problem solver or 'reasoning' component, which provides the functionality for the original task of the software, and the *explainer*. Both, originator and explainer, have their own knowledge container to support their tasks as indicated by the two database symbols. EASD models important aspects for understanding the originator and its application domain. Originator and explainer need to be tightly coupled to help the explainer provide knowledge about the inner workings of the originator.

3 The MACE

The mining and analysis continuum of explaining (MACE) provides different perspectives on the same problem. It considers

1. Different explanation goals
2. Different kinds of explanation
3. Modes of presentation
4. Level of detail of explanation: concrete vs. abstract
5. Utilisation of different knowledge container.
6. Privacy: Which data/information or knowledge from the different knowledge containers is actually revealed to the user?

In the following, we first describe the data mining foundations of the MACE in some detail, before we outline its explanation dimensions, and discuss the issues involved.

3.1 Explanation-Aware Mining and Analysis

In the mining process, explanation features are involved before, during, and after the respective data mining main step, i.e., the *modelling* step. Therefore, we take a broad view regarding the data mining system as the originator, and provide explanation capabilities for each of the datamining steps. In short, the involved mechanisms can be described as follows: The input of the system is given by a (descriptive) specification of the process, the (source) data, and optional background knowledge. The system output is given by a data mining model, e.g., a set of patterns. The output is then accompanied by a "description" of the elementary mining steps, i.e., traces and/or logs of the respective events and steps of the process. The output can then be explained in terms of the input data, additional background knowledge and the intermediate results (trace). Additionally, setting up the specification itself is often a difficult task, for which appropriate explanation features are crucial.

3.2 Explanation Continuum

In the following, we discuss the different dimensions of explanation in the mining and analysis context (Fig. 2), beginning with goals and kinds of explanation, which are immediately useful classifications and a good starting point for explanation-aware application development [12]. In designing a software system knowing about goals and kinds of explanations helps with structuring available knowledge and deciding which knowledge further is required for exhibiting certain explanation capabilities.

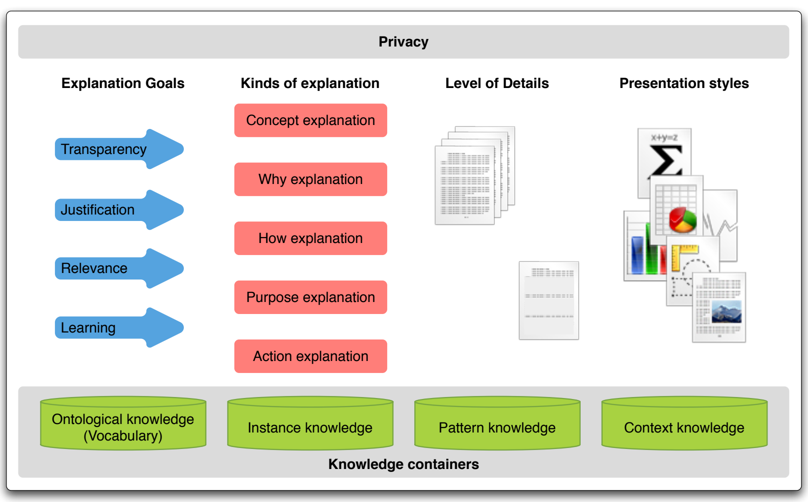

Fig. 2. Overview of the explanation dimensions

Kinds of explanation There are several useful kinds of explanations for knowledge-based systems, i.e., *concept*, *purpose*, *why*, *how*, and *action explanations* [18].

Concept explanations map concepts unknown to the user to already known concepts. The goal of purpose explanations is to describe the purpose of a fact or object. Why explanations justify a fact or an event. How explanations, a special case of why explanations, describe processes in general that lead to an event by providing a causal chain. They provide information about the function of a device. Action explanations explain or predict the behaviour of 'intelligent systems' on the basis of 'known' goals, beliefs, constraints, and rationality assumptions built into the system. Action explanations also answer how questions, but in a concrete context.

The user and/or application goals relate mainly to the type of explanation. During data mining, a data-driven approach starts with the (intermediate/final) results of the mining step. Then, reconstructive explanations [21] are provided by analysing the trace of the system – for why, how, purpose and action explanations. Concept explanations do not necessarily need the analysis of the trace, if they can directly apply the ontological knowledge. Helpful explanation is also provided by adaptation suggestions - comparing the current specification to stored experiences.

Explanation goals help software designers focus on user needs and expectations towards explanations and help to understand what the system has to be able to explain and when to explain something. Sørmo et al. [16, 17] suggest a respective set of explanation goals, i.e., *transparency*, *justification*, *relevance*, and *learning*.

The transparency goal aims at imparting an understanding of how a system found an answer, allowing users to examine the way the system reasons and allows them to look for explanations for why the system has reached a surprising or anomalous result.

The justification goal aims at increasing confidence in the advice or solution offered by a system by giving some kind of support for the conclusion suggested by the system. This goal allows for a simplification of the explanation compared to the actual process the system goes through to find a solution.

An explanation adhering to the relevance goal justifies the strategy pursued by the respective system. This is in contrast to the previous two goals that focus on the solution. The reasoning trace type of explanations may display the strategy of the system implicitly, but it does not argue why it is a good strategy.

Users do not always understand the terms used by a system, especially when starting to use a system. This may be because the user is a novice in the domain, but also because different people can use terms differently or organise knowledge in different ways. It may not be clear, even to an expert, what the system means when using a specific term, and he or she may want to get an explanation of what the system means when using it.

Contextual explanation provides information about the context of the results with respect to already known knowledge, e.g., considering deviations, or contradictions. The transparency of the results can be significantly increased using contextual, why, how, or purpose explanations. The explainer should then show connections between the knowledge containers involved and increase the validity/acceptance of the results using contextual/concept explanation. Also, explanation can be used for improving the modelling of the data mining step, e.g., for documenting and justifying design decisions.

Detail The *detail dimension* concerns the level of detail of the explanation session. Then, the explanation session can make use of various detail abstractions that mostly concern aggregation levels of the data. Abstract explanations, for example, can be enabled using high-level concepts contained in the ontological knowledge container, while finer-grained explanations are provided by applying elements of the low-level presentation dimensions, for example, using textual documents contained in the support knowledge container. Unstructured information can be seen as a very basic form of low-level but very specific and helpful information, consider medical discharge letters or electronic patient records, for example.

Presentation The *presentation dimension* of explaining, i.e., the explanation session information, needs to be performed in an appropriate way, e.g., using textual information, aggregation such as tables or visualisations for more aggregation and abstraction. The design issues involved here are also strongly connected to the *detail dimension*, since the level of detail needs to be reflected by the presentation options and the presentation modes need to be compatible with the detail level.

In the continuum, the presentation dimension provides seamless drill-down/roll-up capabilities similar to OLAP [9] techniques connected with the detail dimension. For instance, information can first be presented using diagrams, then a chunk of knowledge can be inspected using (aggregated) tables, and individual cells can then be analysed on the (input) data, e.g., textual data, or other tables or rows/columns of these.

Containers In [12], Roth-Berghofer and Cassens outline the combination of explanation goals and kinds of explanations in the context of Case-Based Reasoning, and examine the contribution of the four CBR *knowledge containers* for modelling the necessary knowledge [8, 11]. Casting this idea on the field of data mining, the MACE distinguishes the following containers that include explicit knowledge for explaining:

– The *ontological knowledge container* provides *vocabulary*, i.e., the applied concepts of the application context. In the data mining context, this relates to the used attributes. In addition, it contains information about the used concepts, their relations and their properties. In that sense, the vocabulary forms the basis of the other three containers. Ontological knowledge provides that basis for a lot of explanation options cf. [12]. The ontological knowledge is mainly contained in the knowledge base of the *originator* since it also forms the basis of the data mining process.

- The *pattern knowledge container* contains information about the relations, i.e., the mined patterns. Annotations and categorisation for these, for example, such as *known* or *suspected* relations form the basis of explaining the mined patterns in context. The pattern knowledge container has a strong connection to the support knowledge and the ontological knowledge container, since the patterns can always be linked to the ontology and also be supplemented with unstructured information, e.g., by comments or other textual (unstructured) information. The pattern knowledge can both be contained in the *originator* and the *explainer* depending on the application context. If the pattern knowledge is being utilised throughout the data mining session, then the knowledge needs to be embedded into the knowledge base of the originator. On the other hand, if the pattern knowledge is mainly used for enhancing the explanation, e.g., by setting the patterns into context, then the knowledge is usually located in the knowledge base of the explainer.
- The *instance knowledge container*, i.e., the *case base* presents the set of structured cases/instances that are used for data mining. The cases/instances provide the basis for very specific and concrete explanations and require appropriate selection and filtering options. Since the instance knowledge is operational knowledge for the originator, it is included in the respective knowledge base.
- The *context knowledge container* provides additional information about the domain and its concepts, e.g., unstructured information containing the respective concepts. In the medical domain, for example, this could be given by medical records for individual patients. The context knowledge container covers all types for supporting, optimising, or restricting explanations. For the latter case, context knowledge can include information, e.g., about privacy issues of the data, such that certain features of an explanation can be suppressed, if necessary. Since context knowledge is operational knowledge for explaining it is provided by the explanation knowledge base.

Thus, only the context knowledge container includes unstructured information, i.e., textual documents, while all other knowledge container provide structured knowledge for the explanation process.

Privacy Whenever data is collected from heterogeneous sources, aggregation of the data can reveal a lot more information than the single data sources. Consider the shopping domain, for example: In this domain, a lot of information is collected by both the shop and, e.g., a credit card company. All shopping data together can reveal a lot (more) about the life of a single customer, her interests and habits – compared to 'isolated' data. Especially when data from distributed environments and data sources is combined this provides special requirements for privacy-preserving methods, e.g., [1, 20].

Privacy becomes an even more important issue with the availability and use of Linked (Open) Data. The Linked Open Data project is a Semantic Web effort using the Web (i.e., using standards like HTTP, URIs and RDF) to connect related data that was not previously linked together [7]. One of the most visible examples for Linked Open Data is DBPedia, "a community effort to extract structured information from Wikipedia and to make this information available on the Web".[7] Governments and organisations start to provide Linked Data in vast amounts.[8] These data sets will have a strong impact on data mining solutions.

Therefore, the privacy dimension in explaining is a very important and sensitive issue: If the explanation for a user only concerns his data, then the privacy implications are mostly a minor concern. However, whenever other (personal and) privacy-sensitive data is considered during the explanation session, then appropriate measures need to be considered ranging from a privacy-preserving to an open view on the data, information, and knowledge. Appropriate data aggregation or anonymisation, for example, can be used for ensuring the privacy of the collected data.

4 Ubiquitous and Social Contexts

In the following, we describe three application scenarios of the MACE in social software. The first scenario considers the *Social Resource and Metadata Hub ALOE*. Second, we consider the application of the MACE for the social resource sharing system Bibsonomy. Finally, we analyse the explanation continuum in an ubiquitous scenario. In all scenarios, we will revisit the elements of the MACE and provide illustrative examples in the respective application contexts.

In summary, the main benefit of the MACE is its ability to provide more complex and structured explanations at the same time. Consider a video recommendation from your favourite video portal, for instance. In this case, a video is only recommended based on the previously seen videos, however, usually only a simple explanation is provided, e.g., given by one video that is relevant to the recommended video. Utilising the MACE considers more sophisticated explanation options, based on specialised knowledge which could include *interests* or *preferences* in our example. Then, e.g., more advanced justifications can be provided.

4.1 ALOE

The Social Resource and Metadata Hub ALOE[9] is a Web 2.0 resource sharing platform designed, initially, for learning content of arbitrary format [10]. ALOE supports sharing of files and bookmarks, provides tagging, commenting and rating functionalities and offers various search facilities. A group concept enables users to contact and exchange resources with other users that share similar topics of interest. The ALOE system enables users to share content according to personal interests and to be notified of relevant new resources. Whenever users add resources to their portfolio they have to annotate it with a title and tags,

[7] `http://dbpedia.org/About`
[8] `http://esw.w3.org/TaskForces/CommunityProjects/LinkingOpenData/DataSets`
[9] `http://aloe-project.de/`

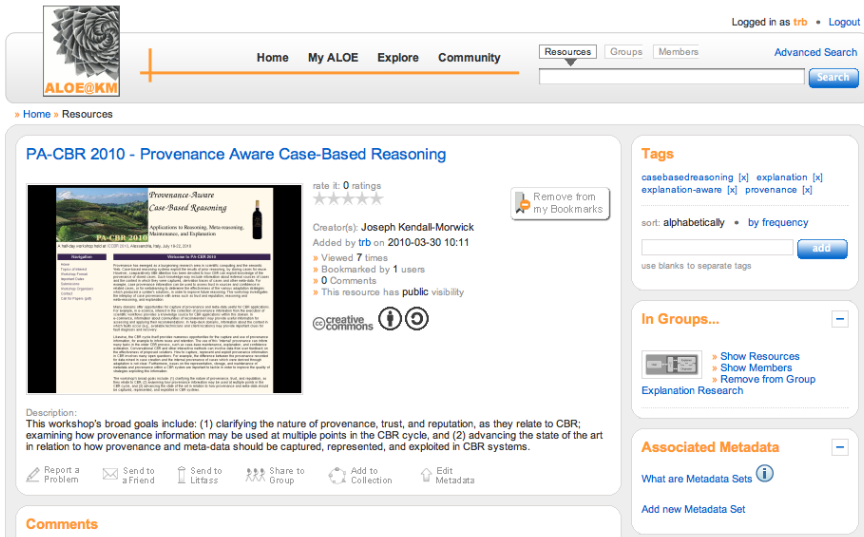

Fig. 3. Screenshot of the ALOE system

supported by information extraction components. Optionally a description as well as author and licensing information may be added. Several deployments of ALOE are in daily use, e.g., at DFKI's Knowledge Management department.

The design of the ALOE platform focused on understandability and ease-of-use. From the beginning transparency was the most important explanation goal to pursue in order to gain the trust of its users. "Concept explanations" and "how" explanations were deemed most suitable for providing transparency in this resource sharing scenario. For Terms that are most likely not self-explanatory concept explanations are provided (indicated by a circled 'i'). For example, in Fig. 3, new users may not know about what metadata sets are. By clicking on the respective link (lower right corner) provides a brief explanation.

C-LINK, a conference organisation system built on top of the ALOE platform is a social sharing tool allowing conference participants to exchange, for instance, resources related to their talks [15]. A content-based recommender system provides event recommendations as well as recommendations of potentially interesting users based on the user's research topics and shared resources. "Why" and "action" explanations help the users to make sense of the recommended resources, events or conference participants.

4.2 BibSonomy

Bibsonomy [6] is a social resource sharing system for managing publications and bookmarks. Besides this basic functionality, Bibsonomy also includes a recommendation framework, user profiles (via a curriculum vitae feature), arbitrary user groups, and links between users implemented by the "friend" functionality.

In this way, Bibsonomy exceeds the typical basic features of a resource sharing system by advanced Web 2.0 and social software principles. However, these features do only complement the resource sharing functionality, which is still at the core of the system.

Explanation features concern the system as the originator of, for example, recommendations, rankings, or "related" user suggestions. Explanations can then be utilised "as is", for enhancing the transparency of the system, but also for adaptations of user settings, and for a general customisation of the (behaviour of) the system.

Concerning the above examples, explanation for recommendations concerns "how" and "why" explanations. This is also related to the ranking features, for which the system should also explain the ranking explicitly via action explanations. Purpose explanations are important for group recommendations, for which concept explanations are also helpful. Finally, user profiles should also be connected to the explanation component, especially if they are involved in the recommendation, or if users are notified about publications of other (interesting) users.

Essentially, the system provides a lot of (social) features and tries to hide these at the same time in order to enable easy access to its features for novice users. Therefore, concept explanations can be applied on the interface meta-level for guiding the user when using the system. Thus, the learning goal can be implemented for enhancing the usability of the system in general. Furthermore, a useful complement for the interface explanations is given by (automatic) adaptive explanations for system changes, e.g., whenever an upgrade of the system is performed.

4.3 UBICON

A broad ubiquitous context is given by the UBICON framework[10], cf. Fig. 4, developed in the context of the VENUS research cluster, that is concerned with the design of social and technological networking issues in situated ubiquitous systems. Therefore, the platform includes ubiquitous and social applications and environments.

First steps of the project are an application using sensor tags for inter-person sensing and localisation, called *Peer Radar* similar to the *Live Social Semantics* approach [2]. The goal of the application is to integrate data from the social semantic web, e.g., from social portals like Facebook, LinkedIn, Delicious, Flickr, lastFM, and from social systems like BibSonomy and KNOWTA in order to integrate those with data from the system, i.e., data about the social interactions and behavioural patterns of the user. Then, context-dependent actions can be applied for a user, i.e., providing helpful information about currently present other users, recommendations for actions, and additional information similar to the paradigm of augmented reality.

[10] `http://www.ubicon.eu`

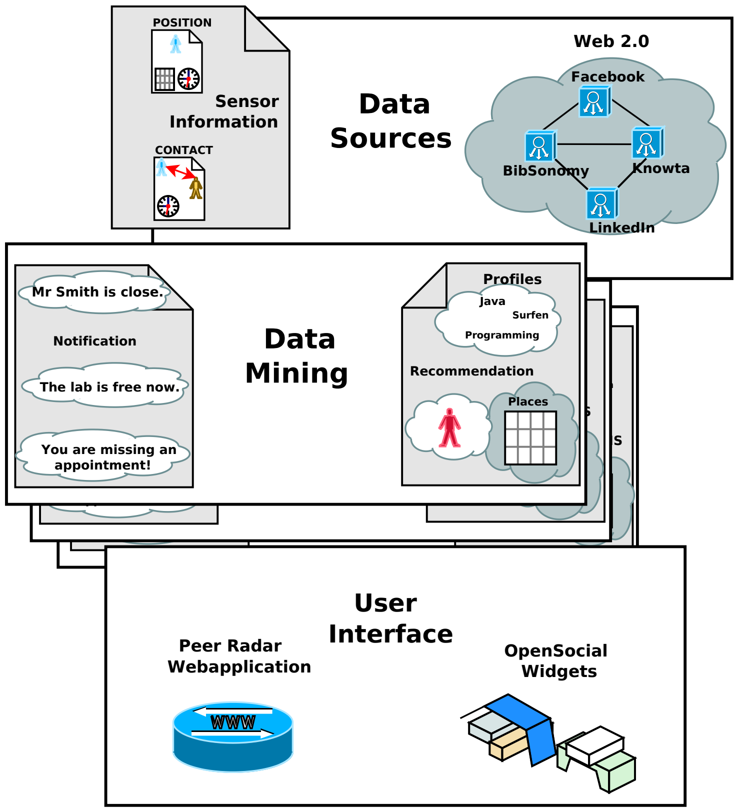

Fig. 4. Overview on Peer Radar System implemented using the UBICON framework for connecting ubiquitous and social environments

In this scenario, the system (as the originator) provides information to the user. In turn it can be explained by the explainer in order to increase the transparency and validity of the information, with the final goal of increasing trust in the system. Suitable recommendations include, for example, *interesting* contacts, topics, context-specific reminders for tasks or persons. These often motivate *actions*, e.g., contacting other users, or working with certain resources. In such cases, the explanation of the proposed mined results is of ultimate importance. In this way, the trust in the actions of the system (and thus in the system itself), can be significantly increased.

Concerning the end-user, the ubiquitous social context provides several examples for explanation, in order to reassess decisions proposed by the system, by asking the system for justifications, for the relevancy of recommendations, and for the transparency of the mining steps. All kinds of explanations can, for example, be embedded into the recommendation explanation task but also in the reminder task. The final goal is to increase the trust in the system which is crucial for the continuous application in the ubiquitous and social context. Therefore, transparent explanations are of high importance.

For constructing the explanations, the explainer can utilise the social structures, social contacts and social knowledge implemented in both the social networks and the social resource sharing system. Using these, complex and structured explanations can be obtained according to the principles of the MACE, e.g., justifications for recommendations according to different details of interested users and/or interest profiles.

5 Conclusions

This paper discussed the application of the continuum of explaining for data mining and analysis in the context of social software: It described how data mining results can be analysed on several continuous dimensions and levels. Then, the mined patterns and data mining models can be inspected in context depending on the level of detail and the concrete perspective. Three application scenarios in ubiquitous and social applications presented exemplary contexts for the presented approach and demonstrated its capabilities.

For future work, we want to investigate ontological explanations in more detail, especially in applications of ubiquitous and social environments. These provide ample options for mining and analysis, enable the inclusion of structured and unstructured data and helpful background knowledge. Furthermore, appropriate tool support is necessary, especially regarding the presentation dimensions. Therefore, we want to investigate advanced explanation-aware presentation techniques, e.g., in the context of the Knowta [3, 4] system and the Ubicon framework, focusing on the concrete explanation-enhancing design issues.

Acknowledgements

This work has been partially supported by the VENUS research cluster at the interdisciplinary Research Center for Information System Design (ITeG) at Kassel University.

References

1. Agrawal, R., Srikant, R.: Privacy-Preserving Data Mining. In: Proc. ACM SIGMOD Conference on Management of Data. pp. 439–450. ACM Press (2000)
2. Alani, H., Szomszor, M., Cattuto, C., den Broeck, W.V., Correndo, G., Barrat, A.: Live Social Semantics. In: International Semantic Web Conference. pp. 698–714 (2009)
3. Atzmueller, M., Haupt, F., Beer, S., Puppe, F.: Knowta: Wiki-Enabled Social Tagging for Collaborative Knowledge and Experience Management. In: Proc. International Workshop on Design, Evaluation and Refinement of Intelligent Systems (DERIS). vol. CEUR-WS (2009)

4. Atzmueller, M., Lemmerich, F., Reutelshoefer, J., Puppe, F.: Wiki-Enabled Semantic Data Mining - Task Design, Evaluation and Refinement. In: Proc. International Workshop on Design, Evaluation and Refinement of Intelligent Systems (DERIS). vol. CEUR-WS (2009)
5. Atzmüller, M., Roth-Berghofer, T.: Ready for the MACE? The Mining and Analysis Continuum of Explaining uncovered. Research Report RR-10-02, Deutsches Forschungszentrum für Künstliche Intelligenz (2010), ISSN 0946-008X
6. Benz, D., Hotho, A., Jäschke, R., Krause, B., Mitzlaff, F., Schmitz, C., Stumme, G.: The Social Bookmark and Publication Management System BibSonomy - A Platform for Evaluating and Demonstrating Web 2.0 Research. VLDB Journal Submitted. (2010)
7. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. International Journal on Semantic Web and Information Systems 5(3), 1–22 (2009)
8. Clancey, W.J.: The Epistemology of a Rule-Based Expert System: A Framework for Explanation. Artificial Intelligence 20, 215–251 (1983)
9. Han, J., Kamber, M.: Data Mining: Concepts and Techniques, 2nd Edition. Morgan Kaufmann, San Francisco, USA (2006)
10. Memmel, M., Schirru, R.: Aloe - a socially aware learning resource and metadata hub. In: Wolpers, M., Klamma, R., Duval, E. (eds.) Proceedings of the EC-TEL 2007 Poster Session at (EC-TEL-2007), September 17-20, Crete, Greece. vol. 280 (2007)
11. Richter, M.M.: The knowledge contained in similarity measures. Invited Talk at the First International Conference on Case-Based Reasoning, ICCBR'95, Sesimbra, Portugal (1995)
12. Roth-Berghofer, T.R., Cassens, J.: Mapping goals and kinds of explanations to the knowledge containers of case-based reasoning systems. In: Muñoz-Avila, H., Ricci, F. (eds.) Case-Based Reasoning Research and Development, ICCBR 2005, Chicago, IL, USA, August 2005, Proceedings. pp. 451–464. No. 3620 in LNAI, Springer Verlag, Heidelberg (2005)
13. Roth-Berghofer, T.R., Richter, M.M.: On explanation. Künstliche Intelligenz 22(2), 5–7 (May 2008)
14. Schank, R.C.: Explanation Patterns: Understanding Mechanically and Creatively. Lawrence Erlbaum Associates, Hillsdale, NJ (1986)
15. Schirru, R., Baumann, S., Memmel, M., Dengel, A.: Extraction of contextualized user interest profiles in social sharing platforms. submitted to Journal of Universal Computer Science
16. Sørmo, F., Cassens, J.: Explanation goals in case-based reasoning. In: Gervás, P., Gupta, K.M. (eds.) Proceedings of the ECCBR 2004 Workshops. pp. 165–174. Departamento de Sistemas Informáticos y Programación, Universidad Complutense Madrid (2004)
17. Sørmo, F., Cassens, J., Aamodt, A.: Explanation in case-based reasoning – perspectives and goals. Artificial Intelligence Review 24(2), 109–143 (2005)
18. Spieker, P.: Natürlichsprachliche Erklärungen in technischen Expertensystemen. Dissertation, University of Kaiserslautern (1991)
19. Svátek, V., Rauch, J., Ralbovský, M.: Ontology-Enhanced Association Mining. In: Semantics, Web and Mining. LNCS, vol. 4289, pp. 163–179 (2005)
20. Vaidya, J., Clifton, C.: Privacy-Preserving Data Mining: Why, How, and When. IEEE Security and Privacy 2, 19–27 (2004)
21. Wick, M.R., Thompson, W.B.: Reconstructive expert system explanation. Artif. Intell. 54(1-2), 33–70 (1992)