Proceedings of the

2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012)

Organised jointly by the
7th European Conference on Technology Enhanced Learning (EC-TEL 2012)

Workshop chairs:
  Nikos Manouselis
  Hendrik Drachsler
  Katrien Verbert
  Olga C. Santos

Saarbrücken (Germany), 18 - 19 September 2012

Table of Contents

Preface

Nikos Manouselis[1], Hendrik Drachsler[2], Katrien Verbert[3], Olga C. Santos[4]

[1]Agro-Know Technologies, 17 Grammou Str., Vrilissia, 15236, Athens, Greece
nikosm@ieee.org
[2]Open Universiteit Nederlands (OUNL). Centre for Learning Sciences and Technologies
(CELSTEC). PO-Box 2960, 6401 DL Heerlen, The Netherlands
Hendrik.Drachsler@ou.nl
[3]KU Leuven, Departement Computerwetenschappen
Celestijnenlaan 200A, B-3001 Leuven, Belgium
katrien.verbert@cs.kuleuven.be
[4]aDeNu Research Group, Artificial Intelligence Department, Computer Science School,
UNED, C/Juan del Rosal, 16. 28040 Madrid, Spain
ocsantos@dia.uned.es

Abstract. The 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012) presents the current status related to the design, development and evaluation of recommender systems in educational settings. It emphasizes the importance of recommender systems for Technology Enhanced Learning (TEL) to support learners with personalized learning resources and suitable peer learners to improve their learning process. 6 full papers and 3 short papers were accepted for publication, and 1 keynote speaker was invited to the workshop.

Keywords: Technology enhanced learning, recommender systems, educational guidance.

Introduction to RecSysTEL 2012

After the successful edition of the 1st Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2010) [1] held in conjunction with the 5th European Conference on Technology Enhanced Learning (EC-TEL) and the 4th ACM Conference on Recommender Systems (RecSys), we have organized this second edition [2] in conjunction with the 7th European Conference on Technology Enhanced Learning (EC-TEL 2012).

Technology enhanced learning (TEL) aims to design, develop and test socio-technical innovations that will support and enhance learning practices of both individuals and organizations, supporting the creation and management of knowledge within organizational settings and communities. It is an application domain that generally addresses all types of technology research and development aiming to support of teaching and learning activities, and considers meta-cognitive and reflective skills such as self-management, self-motivation, and effective informal and

self-regulated learning. Information retrieval is a pivotal activity in TEL, and the deployment of recommender systems has attracted increased interest during the past years as it addresses the information overload problem in TEL scenarios with a low cost approach.

As already confirmed at RecSysTEL 2010, recommendation methods, techniques and systems open an interesting new approach to facilitate and support learning and teaching. There are plenty a resource available on the Web, in terms of digital learning content, services and people resources (e.g. other learners, experts, tutors) that can be used to facilitate teaching and learning tasks. The challenge is to develop, deploy and evaluate systems that provide learners and teachers with meaningful guidance in order to help identify suitable learning resources from a potentially overwhelming variety of choices.

The previous edition of the workshop moved a step forward in this research line, but there is still need for joining the ever increasing number of researchers working on TEL recommenders to share our progress and go further. By using recommendation technology, this workshop contributes to answer this edition EC-TEL research questions, refined as follows:

- How can TEL recommenders support people for the technology-rich workplace after they have left school?
- How can TEL recommenders promote informal and independent learning outside traditional educational settings?
- How can TEL recommenders apply next generation social and mobile technologies to promote informal and responsive learning?

In this context, several questions are being researched around the application of recommender systems in TEL, such as:

- Which are the user tasks that may be supported by recommender systems in TEL settings?
- What should be the focus of recommendation in TEL - resources, people or both?
- What are the requirements for the deployment of recommender systems in a TEL setting?
- What is needed to create a set of public available data sets ranging from formal to non-formal learning settings for TEL recommender systems?
- Can successful recommendation algorithms and systems from other application areas be applied in TEL and what should be the education related requirements taken into account when doing so?
- How to define evaluation criteria for TEL recommender systems?
- How can the success of SIR systems can be evaluated in the context of teaching, learning and/or TEL community building?

Next, we comment on the contributions of the workshop and acknowledge the support received both from organizations and people.

Contributions

The call for papers was disseminated in relevant lists and communities. We received 13 submissions, and each of them was reviewed using a blind refereeing process by 3 members of the Program Committee with expertise from both the RecSys and TEL communities. The reviewing process was carried out using Ginkgo submission system and took into account the following criteria: relevance, sound, organization and readability. In the end, 6 full papers and 3 short papers were accepted. Moreover, *Stefan Dietze* was invited as keynote speaker by the workshop organizers to share his experience with the participants on linked data as a facilitator for TEL recommender systems in research and practice. More specifically, his contribution focuses on providing an overview of most relevant linked data sources and techniques together with a discussion of their potential for the TEL domain in general and TEL recommender systems based on insights from related European projects, including mEducator and LinkedUp [3].

The accepted contributions covered several topics, such as recommendations in learning objects repositories, recommendations in learning scenarios and recommendations of human resources, the consideration of trust and affective issues in the recommendation process and the usage of different data formats in TEL recommenders. Moreover, the recommenders address both the needs of learners and educators.

In particular, the full papers address the following issues. *Cechinel et al.* describe the results of an experiment for automatically generating quality information about learning resources inside repositories in order to pursuit the automatic generation of internal quality information about resources inside repositories [4]. *Paquette et al.* address the problem of competency comparison, providing some heuristics to help match the competencies of users with those involved in task-based scenario components (actors, tasks, resources) and provide a context for recommendation through a learning scenario model and its web-based implementation [5]. *Manouselis et al.* investigate a real life implementation of a multi-criteria recommender system within a Web portal for organic and sustainable education and try to identify the needed adjustments that need to take place in order for it to better match the requirements of its operational environment [6]. *Fazeli et al.* focus on supporting the educators and propose a research approach to take advantage of the social data obtained from monitoring the activities of teachers while they are using a social recommender to find out what are the most suitable resources for their teaching practices [7]. *Koukourikos et al.* propose the introduction of sentiment analysis techniques on user comments regarding an educational resource in order to extract the opinion of a user for the quality of the latter and take into account its quality as perceived by the community before proposing the resource to another user [8]. *Santos and Boticario* discuss the benefits of considering affective issues in educational recommender systems and describe the extension of the Semantic Educational Recommender Systems (SERS) approach, which is characterized by its interoperability with e-learning services, to deal with learners' affective traits in educational scenarios [9].

In turn, the short papers deal with the following topics. *Anjorin et al.* present a framework to support the development of cross-platform recommender systems for

TEL ecosystems and discuss challenges faced, which was effectively applied to develop a cross-platform recommender system in a TEL ecosystem having Moodle as the Learning Management System, Mahara as the Social Networking Service and Ariadne as Learning Object Repository [10]. *Grandbastien et al.* review existing approaches for recommending resources in persona learning environments and describe a novel approach implemented in the OP4L prototype which combines Social Web presence data and semantic web technologies based on an intensive use of ontological models to represent the learning context [11]. *Niemann et al.* focus on the four most commonly used data representations and identify how they can be mapped onto one another to homogenize the usage of formats [12].

Acknowledgements

References

1. RecSysTEL 2010 website: http://adenu.ia.uned.es/workshops/recsystel2010
2. RecSysTEL 2012 website: http://adenu.ia.uned.es/workshops/recsystel2012
3. Dietze, S. Linked Data as facilitator for TEL recommender systems in research & practice. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Draschler, H., Verber, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 7-10.
4. Cechinel, C., da Silva Camargo, S., Ochoa, X., Sicilia, M.A., Sanchez-Alonso, S.. Populating Learning Object Repositories with Hidden Internal Quality Information. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Draschler, H., Verber, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 11-22.
5. Paquette, G., Rogozan, D., Marino, O. Competency Comparison Relations for Recommendation in Technology Enhanced Learning Scenarios. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Draschler, H., Verber, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 23-34.
6. Manouselis, N., Kyrgiazos, G., Stoitsis, G.. Revisiting the Multi-Criteria Recommender System of a Learning Portal. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Drachsler, H., Verbert, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 35-48.
7. Fazeli, S., Drachsler, H., Brouns, F., Sloep, P. A Trust-based Social Recommender for Teachers. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Drachsler, H., Verbert, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 49-60.
8. Koukourikos, A., Stoitsis, G., Karampiperis, P. Sentiment Analysis: A tool for Rating Attribution to Content in Recommender Systems. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Drachsler, H., Verbert, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 61-70.
9. Santos, O.C., Boticario, J.G. Affective issues in Semantic Educational Recommender Systems. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Draschler, H., Verber, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 71-82.
10. Anjorin, M., Dackiewicz, i., Fernandez, A., Rensing, C. A Framework for Cross-Platform Graph-based Recommendations for TEL. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Draschler, H., Verber, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 83-88.
11. Grandbastien, M., Loskovska, S., Nowakowski, S., Jovanovic, J. Using online presence data for recommending human resources in the OP4L project. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Draschler, H., Verber, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 89-94.

12. Niemann, K., Scheffel, M., Wolpers, M. A Comparison of Usage Data Formats for Recommendations in TEL. In Proceedings of the 2nd Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2012). Manouselis, N., Draschler, H., Verber, K., and Santos, O.C. (Eds.). Published by CEUR Workshop Proceedings, 2012, Vol. 896, p. 95-100.

Linked Data as facilitator for TEL recommender systems in research & practice

Stefan Dietze[1]

[1] L3S Research Center, Leibniz University, Hannover, Germany
dietze@l3s.de

Abstract. Personalisation, adaptation and recommendation are central features of TEL environments. In this context, information retrieval techniques are applied as part of TEL recommender systems to filter and deliver learning resources according to user preferences and requirements. However, the suitability and scope of possible recommendations is fundamentally dependent on the quality and quantity of available data, for instance, metadata about TEL resources as well as users. On the other hand, throughout the last years, the Linked Data (LD) movement has succeeded to provide a vast body of well-interlinked and publicly accessible Web data. This in particular includes Linked Data of explicit or implicit educational nature. The potential of LD to facilitate TEL recommender systems research and practice is discussed in this paper. In particular, an overview of most relevant LD sources and techniques is provided, together with a discussion of their potential for the TEL domain in general and TEL recommender systems in particular based on insights from highly related European projects, mEducator and LinkedUp.

Keywords. Linked Data, Education, Semantic Web, Technology-enhanced Learning, Data Consolidation, Data Integration

1   Introduction

As personalisation, adaptation and recommendation are central features of TEL environments, TEL recommender systems apply information retrieval techniques to filter and deliver learning resources according to user preferences and requirements. While the suitability and scope of possible recommendations is fundamentally dependent on the quality and quantity of available data, data about learners, and in particular metadata about TEL resources, the landscape of standards and approaches currently exploited to share and reuse educational data is highly fragmented.

The latter includes, for instance, competing metadata schemas, i.e., general-purpose ones such as Dublin Core[1] or schemas specific to the educational field, like IEEE Learning Object Metadata (LOM) or ADL SCORM[2] but also interface mechanisms such as OAI-PMH[3] or SQI[4]. These technologies are exploited by educational

---

[1] http://dublincore.org/documents/dces/

[2] Advanced Distributed Learning (ADL) SCORM: http://www.adlnet.org

[3] Open Archives Protocol for Metadata Harvesting
http://www.openarchives.org/OAI/openarchivesprotocol.html

resource repository providers to support interoperability. To this end, although a vast amount of educational content and data is shared on the Web in an open way, the integration process is still costly as different learning repositories are isolated from each other and based on different implementation standards [3].

In the past years, TEL research has already widely attempted to exploit Semantic Web technologies in order to solve interoperability issues. However, while the Linked Data (LD) [1] approach has widely established itself as the de-facto standard for sharing data on the Semantic Web, it is still not widely adopted by the TEL community. Linked Data is based on a set of well-established principles and (W3C) standards, e.g. RDF, SPARQL [5] and use of URIs, and aims at facilitating Web-scale data interoperability. Despite the fact that the LD approach has produced an ever growing amount of data sets, schemas and tools available on the Web, its take-up in the area of TEL is still very limited. Thus, LD opens up opportunities to substantially alleviate interoperability issues and to substantially improve quality, quantity and accessibility of TEL data.

2   Challenges

While there is already a large amount of educational data available on the Web via proprietary and/or competing schemas and interface mechanisms, the main challenge for improving impact of TEL recommender systems is to (a) start adopting LD principles and vocabularies while (b) leveraging on existing educational data available on the Web by non-LD compliant means. Following such an approach, major research challenges need to be taken into consideration towards Web-scale interoperability [3]:

- *Integrating distributed data from heterogeneous educational repositories*: educational data and content is usually exposed by heterogeneous services/APIs such as OAI-PMH or SQI. Therefore, interoperability is limited and Web-scale sharing of resources is not widely supported yet.
- *Metadata mediation and transformation*: educational resources and the services exposing those resources are usually described by using distinct, often XML-based schemas and by making use of largely unstructured text and heterogeneous taxonomies. Therefore, schema and data transformation (into RDF) and mapping are important requirements in order to leverage on already existing TEL data.
- *Enrichment and interlinking of unstructured metadata*: existing educational resource metadata is usually provided based on informal and poorly structured data. That is, free text is still widely used for describing educational resources while use of controlled vocabularies is limited and fragmented. Therefore, to allow machine-processing and Web-scale interoperability, educational metadata needs to be enriched, that is transformed into structured and formal descriptions by linking it to widely established LD vocabularies and datasets on the Web.

Our work builds on the hypotheses that Linked Data offers high potential to improve take-up and impact of TEL recommender systems and introduces key past and future

---

[4] Simple Query Interface: http://www.cen-ltso.net/main.aspx?put=859

projects which serve as building blocks towards *Linked Education*[5], i.e. educational data sharing enabled by adoption of Linked Data principles.

3   Towards TEL data integration and exploitation

In particular, we focus on two projects which address the aforementioned challenges by providing innovative approaches towards (a) integration of heterogeneous TEL data (as part of the *mEducator*[6] project) and (b) exploitation of educational open data addressed by the *LinkedUp*[7] project. With respect to (a) we identify a set of principles (see [2][6]) to address the above challenges:

(P1) **Linked Data-principles:** are applied to model and expose metadata of both educational resources and educational services and APIs. In this way, resources are interlinked but also services' description and resources are exposed in a standardized and accessible way.

(P2) **Services integration:** Existing heterogeneous and distributed learning repositories, i.e. their Web interfaces (services) are integrated on the fly by reasoning and processing of LD-based service semantics (see P1).

(P3) **Schema matching:** metadata retrieved from heterogeneous Web repositories, for instance is automatically lifted into RDF, aligned with competing metadata schemas and exposed as LD accessible via de-referenceable URIs.

(P4) **Data interlinking, clustering and enrichment:** Automated enrichment and clustering mechanisms are exploited in order to interlink data produced by (P3) with existing datasets as part of the LD cloud.

While this work aims at increasing the quantity, quality and accessibility of available educational data on the Web, LinkedUp addresses (b) by aiming to push forward the exploitation of the vast amounts of public, open data available on the Web, in particular by educational institutions and organizations. This will be achieved by identifying and supporting highly innovative large-scale Web information management applications through an open competition (the *LinkedUp Challenge*) and dedicated evaluation framework. The vision of the LinkedUp Challenge is to realise personalised university degree-level education of global impact based on open Web data and information. Drawing on the diversity of Web information relevant to education, ranging from OER metadata to the vast body of knowledge offered by the LD approach, this aim requires overcoming substantial challenges related to Web-scale data and information management involving Big Data, such as performance and scalability, interoperability, multilinguality and heterogeneity problems, to offer personalised and accessible education services. Therefore, the LinkedUp Challenge provides a focused scenario to derive challenging requirements, evaluation criteria, benchmarks and thresholds which are reflected in the LinkedUp evaluation

---

[5] http://linkededucation.org: an open platform to share results focused on educational LD. Long-term goal is to establish links and unified APIs and endpoints to educational datasets.

[6] http://www.meducator.net

[7] LinkedUp: Linking Web Data for Education Project – Open Challenge in Web-scale Data Integration (http://www.linkedup-project.eu)

framework. Information management solutions have to apply data and learning analytics methods to provide highly personalised and context-aware views on heterogeneous Web data. Building on the strong alliance of institutions with expertise in areas such as open Web data management, data integration and Web-based education, key outcomes of LinkedUp include a general-purpose evaluation framework for Web-data driven applications, a set of quality-assured educational datasets, innovative applications of large-scale Web information management, community-building and clustering crossing public and private sectors and substantial technology transfer of highly innovative Web information management technologies.

4   Conclusions

We provided an overview of two efforts both aiming at the overall goal of fostering the reuse of open educational data on the Web. While the accessibility of large-scale amounts of data is a foundation for TEL recommender systems, both efforts contribute to improvements in scope, quantity and quality of recommendations in TEL environments. This includes both, TEL recommender systems in research, where data is required for evaluation and benchmarking, as well as in practice, where data is a core requirement for offering suitable recommendations to users.

Acknowledgments

References

[1] Bizer, C., T. Heath, Berners-Lee, T. (2009). Linked data - The Story So Far. Special Issue on Linked data, International Journal on Semantic Web and Information Systems.

[2] Dietze, S., Yu, H. Q., Giordano, D., Kaldoudi, E., Dovrolis, N. and Taibi, D. (2012), "Linked Education: interlinking educational Resources and the Web of Data", *Proceedings of the 27th ACM Symposium On Applied Computing (SAC-2012), Special Track on Semantic Web and Applications*, Riva del Garda (Trento), Italy, 2012.

[3] Dietze, S., Sanchez-Alonso, S., Ebner, H., Yu, H., Giordano, D., Marenzi, I., Pereira Nunes, B. (2013) Interlinking educational Resources and the Web of Data – a Survey of Challenges and Approaches, accepted for publication in Emerald Program: electronic Library and Information Systems, Volume 47, Issue 1 (2013).

[4] IEEE (2002), "IEEE Standard for Learning Object Metadata", *IEEE Std 1484.12.1-2002*, pp.i–32. doi: 10.1109/IEEESTD.2002.94128.

[5] World Wide Web Consortium (2008). *W3C Recommendation, SPARQL query language for RDF*, 2008, available at:www.w3.org/TR/rdf-sparql-query/

[6] Yu, H. Q., Dietze, S., Li, N., Pedrinaci, C., Taibi, D., Dovrolls, N., Stefanut, T., Kaldoudi, E. and Domingue, J. (2011), "A linked data-driven & service-oriented architecture for sharing educational resources", in *Linked Learning 2011,* Proceedings of the 1st International Workshop on eLearning Approaches for Linked Data Age, May 29, 2011, Heraklion, Greece.

Populating Learning Object Repositories with Hidden Internal Quality Information

Cristian Cechinel[1], Sandro da Silva Camargo[1], Xavier Ochoa[2], Salvador Sánchez-Alonso[3], Miguel-Ángel Sicilia[3]

[1] Computer Engineering Course
Federal University of Pampa, Caixa Postal 07
96400-970, Bagé (RS), Brazil
contato@cristiancechinel.pro.br, camargo.sandro@gmail.com

[2] Escuela Superior Politécnica del Litoral, Campus Gustavo Galindo, Km. 30. Vía Perimetral, Guayaquil, Ecuador, xavier@cti.espol.edu.ec

[3] Information Engineering Research Unit
Computer Science Dept., University of Alcalá
Ctra. Barcelona km. 33.6 – 28871 Alcalá de Henares (Madrid), Spain
salvador.sanchez, msicilia@uah.es

Abstract. It is known that current Learning Object Repositories adopt strategies for quality assessment of their resources that rely on the impressions of quality given by the members of the repository community. Although this strategy can be considered effective at some extent, the number of resources inside repositories tends to increase more rapidly than the number of evaluations given by this community, thus leaving several resources of the repository without any quality assessment. The present work describes the results of an experiment for automatically generate quality information about learning resources inside repositories through the use of Artificial Neural Networks models. We were able to generate models for classifying resources between *good* and *not-good* with accuracies that vary from 50% to 80% depending on the given subset. The preliminary results found here point out the feasibility of such approach and can be used as a starting point for the pursuit of automatically generation of internal quality information about resources inside repositories.

Keywords: Ranking mechanisms; ratings; learning objects; learning object repositories; MERLOT; Artificial Neural Networks

1   Introduction

Current Learning Object Repositories (LORs) normally adopt strategies for the establishment of quality of their resources that rely on the impressions of usage and

evaluations given by the members of the repository community (ratings, tags, comments, likes, lenses). All this information together constitute a collective body of knowledge that further serves as an external memory that can help other individuals to find resources according to their individual needs. Inside LORs, this kind of evaluative metadata (Vuorikari, Manouselis, & Duval, 2008) is also used by search and retrieval mechanisms for properly ranking and recommending resources to the community of users of the repository.

Although such strategies can be considered effective at some extent, the amount of resources inside repositories is rapidly growing every day (Ochoa & Duval, 2009) and it became impractical to rely only on human effort for such a task. For instance, on a quick look at the summary of MERLOT's recent activities, it is possible to observe that in a short period of one month (from May 21th to June 21th), the amount of new resources catalogued in the repository was 9 times more than the amount of new ratings given by experts (peer-reviewers), 6 times more than the amount of new comments (and users ratings) and 3 times more than the amount of new bookmarks (personal collections). This situation of leaving many resources of the current repositories without any measure of quality at all (and consequently unable or at least on a very disadvantage position to compete for a good place during the process of search and retrieval) has raised the concern for the development of new automated techniques and tools that could be used to complement existing manual approaches. On that direction, Ochoa and Duval (2008) developed a set of metrics for ranking the results of learning objects search according to three dimensions of relevance (topical, personal and situational) and by using information obtained from the learning objects metadata, from the user queries, and from other external sources such as the records of historical usage of the resources. The authors contrasted the performance of their approach against the text-based ranking traditional methods and have found significant improvements in the final ranking results. Moreover, Sanz-Rodriguez, Dodero, and Sánchez-Alonso (2010) proposed to integrate several distinct quality indicators of learning objects of MERLOT along with their usage information into one overall quality indicator that can be used to facilitate the ranking of learning objects.

These mentioned approaches for automatically measure quality (or calculate relevance) according to specific dimensions depend on the existence and availability of metadata attached to the resources (or inside the repositories), or on measures of popularity about the resources that are obtained only when the resource is publicly available after a certain period of time. As metadata may be incomplete/inaccurate and these measures of popularity will be available just for "old" resources, we propose to apply an alternative approach for this problem. The main idea is to identify intrinsic measures of the resources (i.e., features that can be calculated directly from the resources) that are associated to quality and that can be used in the process of creating models for automated quality assessment. In fact, this approach was recently tested by Cechinel, Sánchez-Alonso, and García-Barriocanal (2011) who developed highly-rated profiles of learning objects available in the MERLOT repository, and have generated Linear Discriminant Analysis (LDA) models based on 13 learning objects intrinsic features. The generated models were able to classify

resources between good and not-good with 72.16% of accuracy, and between good and poor with 91.49% of accuracy. Among other things, the authors have concluded that highly-rated learning objects profiles should be developed taking into consideration the many possible intersections among the different disciplines and types of materials available in the MERLOT repository, as well as the group of evaluators who rated the resources (whether they are formed by experts or by the community of users). For instance, the mentioned models were created for materials of *Simulation* type belonging to the discipline of *Science & Technology*, and considering the perspective of the peer-reviewers ratings. On an another round of experiments, Cechinel (2012) also tested the creation of automated models through the creation of statistical profiles and the further use of data mining classification algorithms for three distinct subsets of MERLOT materials. On these studies the author were able to generate models with good overall precision rates (up to 89%) but the author highlighted that the feasibility of the models will depend on the specific method used to generate them, the specifics subsets to which they are being generated for, and the classes of quality included in the dataset. Moreover, the models were generated by using considerably small datasets (around 90 resources each), and were evaluated using the training dataset, i.e., the entire dataset was used for training and for evaluating.

The present work expands the previous works developed by Cechinel (2012) and Cechinel et al. (2011) by generating and evaluating models for automated quality assessment of learning objects stored on MERLOT focusing on populating the repository with hidden internal quality information that can be further used by ranking mechanisms. On the previous works, the authors explored the creation of statistical profiles of highly-rated learning objects by contrasting information from the *good* and *not-good* resources and then used these profiles to generate models for quality assessment. In the present work we are testing a slightly different and more algorithmic approach, i.e., the models here are being generated exclusively through the use of data mining algorithms. Moreover, we are also working with a larger collection of resources and a considerably higher number of MERLOT subsets. The rest of this paper is structured as follows. Section 2 presents existing research focused on identifying intrinsic quality features of resources. Section 3 describes the methodology followed for the study and section 4 discusses the results found. Finally, conclusions and outlook are provided in Section 5.

2 Background

From our knowledge, besides the recent work of Cechinel et al. (2011), there is still no empirical evidence of intrinsic metrics that could serve as indicators of quality for LOs. However, there are some works in adjacent fields which can serve us as a source of inspiration. For instance, empirical evidence of relations from intrinsic information and other characteristics of LOs have been found in (Meyer, Hannappel, Rensing, & Steinmetz, 2007), where the authors developed a model for classifying the didactic functions of a learning object based on measures about the length of the

text, the presence of interactivity and information contained in the HTML code (lists, forms, input elements). Mendes, Hall, and Harrison (1998) have identified evidence in some measures to evaluate sustainability and reusability of educational hypermedia applications, such as, the type of link, and the structure and size of the application. Blumenstock (2008) has found the length of an article (measured in words) as a predictor of quality in Wikipedia. Moreover, Stvilia, Twidale, Smith & Gasser (2005) have been able to automatically discriminate high quality articles voted by the community of users from the rest of the articles of the collection. In order to do that, the authors developed profiles by contrasting metrics of articles featured as best articles by Wikipedia editors against a random set. The metrics were based on measures of the article edit history (total number of edits, number of anonymous user edits, for instance) and on the article attributes and surface features (number of internal broken links, number of internal links, number of images, for instance). At last, in the field of usability, Ivory and Hearst (2002) have found that good websites contain (for instance) more words and links than the regular and bad ones.

Our approach is initially related exclusively to those aspects of learning objects that are displayed to the users and that are normally associated to the dimensions of presentation design and interaction usability included in LORI (Nesbit, Belfer, & Leacock, 2003) and the dimension of information quality (normally mentioned in the context of educational digital libraries). Precisely, the references for quality assurance used in here are the ratings given by the peer-reviewers (experts) of the repository.

3   Methodology

The main objective of this research was to obtain models that could automatically identify *good* and *not-good* learning objects inside repositories based on the intrinsic features of the resources. The methodology that we followed was the development of models though the use of data mining algorithms over information of learning objects catalogued on MERLOT repository. For that, a database was collected from the repository and qualitative classes of quality of *good* and *not-good* were generated considering the terciles of the ratings of the resources. These classes of quality were then used as the reference output for the generation of the models.

3.1   Data Collection

A database was collected from MERLOT through the use of a crawlerthat systematically traversed the pages and collected information related to 35 metrics of the resources. The decision of choosing MERLOT lays mainly on the fact that MERLOT has one of the largest amount of registered resources and users, and it implements a system for quality assurance that works with evaluations given by experts and users of the repository. Such system can serve as baseline for the creation of the learning object classes of quality. As MERLOT repository is mainly formed by

learning resources in the form of websites, we evaluated intrinsic metrics that are supposed to appear in such technical type of material (i.e., link measures, text measures, graphic measures and site architecture measures). The metrics collected for this study (see Table 1) are the same as used by Cechinel et al. (2011) and some of them have also been mentioned in other works which tackled the problem of assessing quality of resources (previously presented in section 2).

Table 1: Metrics collected for the study

Class of Measure	*Metric*
Link Measures	Number of Links, Number of Unique[a] Links, Number of Internal Links[b], Number of Unique Internal Links, Number of External Links, Number of Unique External Links
Text Measures	Number of Words, Number of words that are links[c]
Graphic, Interactive and Multimedia Measures	Number of Images, Total Size of the Images (in bytes), Number of Scripts, Number of Applets, Number of Audio Files, Number of Video Files, Number of Multimedia Files
Site Architecture Measures	Size of the Page (in bytes), Number of Files for downloading, Total Number of Pages

[a] The term Unique stands for "non-repeated"
[b] The term internal refers to those links which are located at some directory below the root site
[c] For these metrics the average was not computed or does not exist

As resources in MERLOT vary considerably in size, a limit of 2 levels of depth was established for the crawler, i.e., metrics were computed for the root node (level 0 - the home-page of the resource), as well as for the pages linked by the root node (level 1), and for the pages linked by the pages of the level 1 (level 2[1]). As it is shown in table 1, some of the metrics refer to the total sum of the occurrences of a given attribute considering the whole resource, and other metrics refer to the average of this sum considering the number of the pages computed. For instance, an object composed by 3 pages and containing a total of 30 images, will have a total number of images of 30, and an average number of images equals to 10 (30/3). Information of a total of 20,582 learning resources was collected. From this amount, only 2,076 were peer-reviewed, and 5 of them did not have metadata regarding the category of discipline or the type of material and were disregarded. Considering that many subsets are formed by very small amount of resources, we restrained our experiment to just a few of them. Precisely, we worked with 21 subsets formed by the following types of material: *Collection*, *Reference Material*, *Simulation* and *Tutorial*, and that had 40 resources or more[2]. In total, we worked with information of 1,429 learning resources which represent 69% of the total collected data. Table 2 presents the frequency of the materials for each subset used in this study.

---

[1] Although this limitation may affect the results, the process of collecting the information is extremely slow and such limitation was needed. In order to acquire the sample used in this study, the crawler kept running uninterruptedly for 4 full months.

[2] The difficulties for training, validating and testing predictive models for subsets with less than 40 resources would be more severe.

Table 2: Frequency of materials for the subsets used in this study (intersection of category of discipline and material type)

Material Type/Discipline	Arts	Business	Education	Humanities
Collection		52	56	43
Reference Material		83	40	51
Simulation	57	63	40	78
Tutorial		76	73	93

Material Type/Discipline	Mathematics and Statistics	Science & Technology	Social Sciences
Collection	50	80	
Reference Material	68	102	
Simulation	40	150	
Tutorial	48	86	

3.2   Classes of Quality

As the peer-reviewers ratings tend to concentrate above the intermediary rating 3, classes of quality were created using the terciles of the ratings for each subset[3]. Resources with ratings below the first tercile are classified as *poor*, resources with ratings equal or higher the first tercile and lower than the second tercile are classified as *average*, and resources with ratings equal or higher the second tercile are classified as *good*. The classes of quality *average* and *poor* were then joined in another class called *not-good*.

3.3   Mining models for automated quality classification of learning objects

The classes of quality were used as the output reference for generating and testing models for automated quality assessment of the resources through the use of Artificial Neural Networks (ANNs). The choice of using ANNs rests on the fact that they are adaptive, distributed, and highly parallel systems which have been used in many knowledge areas and have proven to solve problems that require pattern recognition (Bishop, 2006). Moreover, ANNs are among the types of models that have also shown good accuracies on the previous works mentioned before. Finally, we have initially tested other approaches (with rules and trees) and they presented maximum accuracies around 60%. As ANNs presented the best preliminary results we selected this approach for the present study.

The experiments were conducted with the Neural Network toolbox of Matlab. For each subset we randomly selected 70% of the data for training, 15% for testing and 15% for validation, as suggested by Xu, Hoos, and Leyton-Brown (2007). We tested the Marquardt –Levenberg algorithm (Hagan & Menhaj, 1994) using from 1 to 30 neurons in all tests. In order to obtain more statistically significant results (due to the small size of the data samples), each test was repeated 10 times and the average results were computed. The models were generated to classify resources between *good* and *not-good*.

---

[3] The terciles of each subset were omitted from the paper due to a lack of space

4   Results and Discussion

The models presented different results depending on the subset used for training. Most of the models tend to classify *not-good* resources better than *good* ones which can probably be a result of the uneven amount of resources of each class inside the datasets (normally formed by 2/3 of *not-good* and 1/3 of *good*). These tendencies can be observed in figure 2[4].

The number of neurons used on the construction of the models has different influences depending on the subsets. A Spearman's rank correlation (r_s) analysis was carried out to evaluate whether there are associations between the number of neurons and the accuracies achieved by the models. This test serves to the purpose of observing the pattern expressed by the models on predicting quality for the given subsets. For instance, assuming *x* as a predictive model for a given subset *A*, and *y* as a predictive model for a given subset *B*; if *x* has less neurons than *y* and both have the same accuracies, the patterns expressed in *A* are simpler than the ones expressed in *B*. This means to say that it is easier to understand what is *good* (or *not-good*) in the subset *A*. Table 3 shows the results of such analysis.

In Table 3 (-) stands for no association between the number of neurons and the accuracy of the model for classifying a given class, (↑) stands for a positive association, and (↓) stands for a negative association. The analyses considered a 95% level of significance. As it can be seen in the table, the number of neurons influences on the accuracies for some classes of quality of some subsets. For instance, the number of neurons presents a positive association with the accuracies for classifying *good* resources in the 6 (six) following subsets: *Business ∩ Simulation*, *Business ∩ Tutorial*, *Education ∩ Collection*, *Education ∩ Tutorial*, *Humanities ∩ Tutorial*, and *Science & Technology ∩ Simulation*. Moreover, the number of neurons presents a negative association with the accuracies for classifying *not-good* resources in the 8 (eight) following subsets: *Arts ∩ Simulation*, *Business ∩ Tutorial*, *Education ∩ Collection*, *Education ∩ Simulation*, *Education ∩ Tutorial*, *Education ∩ Humanities*, *Science & Technology ∩ Simulation*, and *Science & Technology ∩ Tutorial*. Finally, there are no positive associations between the number of neurons and the accuracies for classifying *not-good* resources; neither there are negative associations between the number of neurons and the accuracies for classifying *good* resources.

---
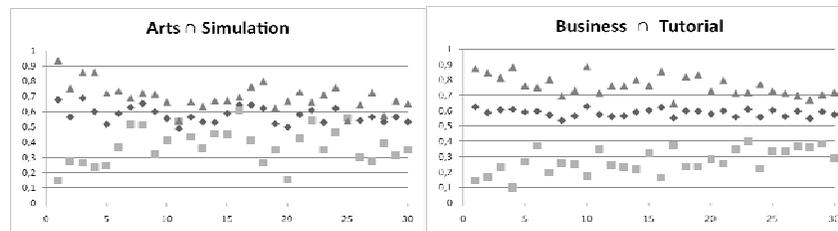
[4] Just some models were presented in the figure due to a lack of space
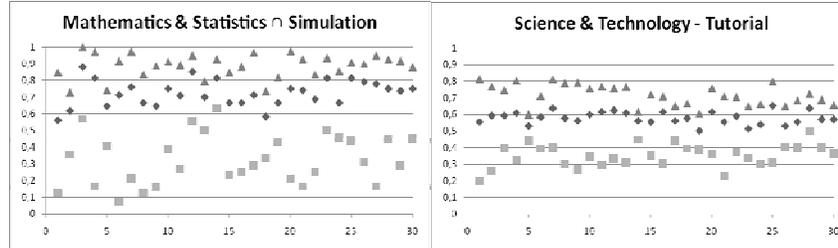
Fig.2. Accuracies of the some models versus number of neurons. Overall accuracies (lozenges), accuracies for the classification of *good* resources (squares) and *not-good* resources (triangles)

Table 3: Tendencies of the accuracies according to the number of neurons used for training (*good | not-good*)

Subset	Arts	Business	Education	Humanities	Math & Statistics	Science & Tech
Collection		- | -	↑ | ↓	- | -	- | -	- | -
Reference Material		- | -	- | -	- | ↓	- | -	- | -
Simulation	- | ↓	↑ | -	- | ↓	- | -	- | -	↑ | ↓
Tutorial		↑ | ↓	↑ | ↓	↑ | -	- | -	- | ↓

In order to evaluate how to select the best models for quality assessment, it is necessary to understand the behavior of the models for classifying both classes of quality included on the datasets. Considering that, a Spearman's rank correlation (r_s) analysis was also carried out to evaluate whether there are associations between the accuracies of the models for classifying *good* and *not-good* resources. Such analysis serves to evaluate the trade-offs of selecting or not a given model for the present purpose. Most of the models have presented strong negative correlations between the accuracies for classifying *good* and *not-good* resources. The results of both analyses suggest that the decision of selecting a model for predicting quality must take into account that, as the accuracy for classifying resources from one class increases, the accuracy for classifying resources of the other class decreases. Considering that, the question lies on establishing which would be the cutting point for acceptable accuracies so that the models could be used for our purpose. In other words, it is necessary to establish the minimum accuracies (cutting point) that the models must present for classifying both classes (*good* and *not-good*) so that they can be used for generating hidden quality information for the repository.

For the present study, we are considering that the models must present accuracies higher than 50% for the correct classification of *good* and *not-good* resources (simultaneously) in order to be considered as useful. It is known that the decision of selecting the minimum accuracies for considering a model as efficient or not will depend on the specific scenario/problem for which the models are being developed for. Here we are considering that accuracies higher than 50% are better than the merely random.

Table 4 presents the top-2 models for each subset considering their overall accuracies, and their accuracies for classifying *good* and *not-good* resources (ordered by the accuracy for classifying *good* resources).

Table 4: Two best models for each subset (ordered by the accuracies for classifying *good* resources)

Subset	N	OA	G	NG	Subset	N	OA	G	NG
Arts ∩ Simulation	16	0,65	0,61	0,70	Business ∩ Collection	11	0,56	0,61	0,60
	25	0,55	0,56	0,54		25	0,57	0,60	0,59
Business ∩ Reference	8	0,58	0,54	0,59	Business ∩ Simulation	24	0,64	0,67	0,60
	5	0,59	0,53	0,68		30	0,57	0,62	0,55
Business ∩ Tutorial	23	0,61	0,40	0,72	Education ∩ Collection	26	0,51	0,6	0,49
	29	0,59	0,38	0,71		29	0,51	0,6	0,44
Education ∩ Reference	16	0,60	0,63	0,70	Education ∩ Simulation	20	0,52	0,62	0,5
	20	0,58	0,54	0,71		12	0,53	0,59	0,56
Education ∩ Tutorial	27	0,47	0,49	0,47	Humanities ∩ Collection	14	0,6	0,75	0,51
	29	0,53	0,43	0,61		19	0,63	0,69	0,68
Humanities ∩ Reference Mat.	29	0,47	0,59	0,49	Humanities ∩ Simulation	4	0,69	0,76	0,69
	10	0,58	0,5	0,65		9	0,79	0,75	0,79
Humanities ∩ Tutorial	25	0,56	0,60	0,58	Math.& Statistics ∩ Collection	28	0,5	0,61	0,54
	21	0,51	0,59	0,54		27	0,49	0,57	0,46
Math. ∩ Reference Mat.	22	0,63	0,54	0,72	Math.& Statistics ∩ Simulation	14	0,81	0,63	0,93
	18	0,53	0,48	0,60		3	0,88	0,57	1
Mathematics ∩ Tutorial	26	0,69	0,79	0,64	Science & Tech. ∩ Collection	17	0,58	0,60	0,54
	25	0,70	0,77	0,61		3	0,56	0,54	0,60
Science & Tech. ∩ Reference Mat.	19	0,59	0,63	0,56	Science & Tech. ∩ Simulation	29	0,57	0,58	0,61
	16	0,55	0,58	0,58		19	0,58	0,52	0,62
Science & Tech. ∩ Tutorial	28	0,64	0,50	0,72					
	14	0,56	0,45	0,61					

In table 4, N stands for the number of neurons in the model, OA stands for the overall accuracy, G for the accuracy for classifying good resources and NG for the accuracy for classifying not-good resources. As it can be seen in the table, and considering the established minimum cutting-point, it was possible to generate models for almost all subsets. From the 42 models presented in the table, only 10 did not reach the minimum accuracies (white in the table). Moreover, 22 of them presented accuracies between 50% and 59.90% (gray hashed in the table), and 9 presented both accuracies higher than 60% (black hashed in the table). We have also found 1 (one) model with accuracies higher than 70% (for *Humanities ∩ Simulation*). The only three subsets to which the models did not reach the minimum accuracies were: *Business ∩ Tutorial*, *Education ∩ Collection* and *Education ∩ Tutorial*. On the other hand, the best results were found for: *Humanities ∩ Simulation, Mathematics ∩ Tutorial, Humanities ∩ Collection, Business ∩ Simulation*, *Arts ∩ Simulation* and

Business ∩ *Collection.* One of the possible reasons why it was not feasible to generate good models for all subsets may rest on the fact that the real features associated to quality on those given subsets might not have been collected by the crawler.

In order to select the most suitable model one should take into consideration that the model's output is going to be used as information during the ranking process, and to evaluate the advantages and drawbacks of a lower accuracy for classifying *good* resources in contraposition to a lower accuracy for classifying *not-good* resources. The less damaging situation seems to occur when the model classify as *not-good* a *good* material. In this case, *good* materials would just remain hidden in the repository, i.e., in bad ranked positions (a similar situation to the one of not using the models). On the other hand, if the model classifies as *good* a resource that is *not-good*, it is most likely that this resource will be put at a higher rank position, thus increasing its chances of being accessed by the users. This would mislead the user towards the selection of a "not-so-good" quality resource, and it could put in discredit the ranking mechanism.

5 Conclusions and Outlook

It is known that LORs normally use evaluative information to rank resources during the process of search and retrieval. However, the amount of resources inside LORs increases more rapidly than the number of contributions given by the community of users and experts. Because of that, many LOs that do not have any quality evaluation receive bad rank positions even if they are of high-quality, thus remaining unused (or unseen) inside the repository until someone decides to evaluate it. The models developed here could be used to provide internal quality information for those LOs still not evaluated, thus helping the repository in the stage of offering resources. Among other good results, one can mention the model for *Humanities* ∩ *Simulation* that is able to classify *good* resources with 75% of precision and *not-good* resources with 79%; and the model developed for *Mathematics* ∩ *Tutorial* with 79% of precision for classifying *good* resources and 64% for classifying *not-good* ones. As the models would be used inside repository and the classifications would serve just as input information for searching mechanisms, it is not necessarily required that the models provide explanations about their reasoning. Models constituted of neural networks (as the one tested in the present study) can perfectly be used in such a scenario.

Resources recently added to the repository would be highly benefited by such models since that they hardly receive any assessment just after their inclusion. Once the resource finally receives a formal evaluation from the community of the repository, the initial implicit quality information provided by the model could be disregarded. Moreover, this "real" rating could be used as feedback information so that the efficiency of the models could be analyzed, i.e. to evaluate whether or not the users agree with the models decisions.

Future work will try to include more metrics still not implemented, such as, for instance, the number of colors and different font styles, the existence of adds, the number of redundant and broken links, and some readability measures (e.g. Gunning Fog index and Flesch-Kincaid grade level). Besides, as pointed out by Cechinel and Sánchez-Alonso (2011), both communities of evaluators in MERLOT (users and peer-reviewers) are communicating different views regarding the quality of the learning objects refereed in the repository. The models tested here are related to the perspective of quality given by peer-reviewers. Future work will test models created with the ratings given by the community of users and compare their performances with the present study. Moreover, as the present work is context sensitive, it is important to evaluate whether this approach can be extended to other repositories. As not all repositories adopt the same kind of quality assurance that MERLOT does, alternative quality measures for contrasting classes between *good* and *not-good* resources must be found. Another interesting possible direction is to classify learning resources according to their granularity, and use this information as one of the metrics to be evaluated during the creation of the highly-rated profiles. At last, we could use the values calculated by the models for all the resources and compare the ranking of MERLOT with the ranking performed through the use of these "artificial" quality information.

It is important to mention that the present approach does not intend to replace traditional evaluation methods, but complement them providing a useful and inexpensive quality assessment that can be used by the repositories before more time and effort consuming evaluation is performed.

Acknowledgments

References

Bishop, Christopher M. (2006). *Pattern Recognition and Machine Learning*: Springer.

Blumenstock, Joshua E. (2008). *Size matters: word count as a measure of quality on wikipedia*. Paper presented at the Proceedings of the 17th international conference on World Wide Web, Beijing, China.

Cechinel, Cristian. (2012). *Empirical Foundations for Automated Quality Assessment of Learning Objects inside Repositories.* (Ph.D. Doctoral Thesis), University of Alcalá, Alcalá de Henares.

Cechinel, Cristian, & Sánchez-Alonso, Salvador. (2011). Analyzing Associations between the Different Ratings Dimensions of the MERLOT Repository. *Interdisciplinary Journal of E-Learning and Learning Objects 7*, 1-9.

Cechinel, Cristian, Sánchez-Alonso, Salvador, & García-Barriocanal, Elena. (2011). Statistical profiles of highly-rated learning objects. *Computers & Education, 57*(1), 1255-1269. doi: 10.1016/j.compedu.2011.01.012

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *Neural Networks, IEEE Transactions on, 5*(6), 989-993. doi: 10.1109/72.329697

Ivory, M. Y., & Hearst, M. A. (2002). *Statistical profiles of highly-rated web sites*. Paper presented at the Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves, Minneapolis, Minnesota, USA.

Mendes, Emilia, Hall, Wendy, & Harrison, Rachel. (1998). Applying Metrics to the Evaluation of Educational Hypermedia Applications. *Journal of Universal Computer Science, 4*(4), 382-403. doi: 10.3217/jucs-004-04-0382

Meyer, Marek, Hannappel, Alexander, Rensing, Christoph, & Steinmetz, Ralf. (2007). *Automatic classification of didactic functions of e-learning resources*. Paper presented at the Proceedings of the 15th international conference on Multimedia, Augsburg, Germany.

Nesbit, John C., Belfer, Karen, & Leacock, Tracey. (2003). Learning object review instrument (LORI). E-learning research and assessment network. Retrieved from http://www.elera.net/eLera/Home/Articles/LORI%20manual.

Ochoa, Xavier, & Duval, Erik. (2008). Relevance Ranking Metrics for Learning Objects. *Learning Technologies, IEEE Transactions on, 1*(1), 34-48. doi: http://dx.doi.org/10.1109/TLT.2008.1

Ochoa, Xavier, & Duval, Erik. (2009). Quantitative Analysis of Learning Object Repositories. *Learning Technologies, IEEE Transactions on, 2*(3), 226-238.

Sanz-Rodriguez, Javier, Dodero, Juan, & Sánchez-Alonso, Salvador. (2010). Ranking Learning Objects through Integration of Different Quality Indicators. *IEEE Transactions on Learning Technologies, 3*(4), 358 - 363. doi: 10.1109/TLT.2010.23

Stvilia, B., Twidale, M. B., Smith, L. C., & Gasser, L. (2005). *Assessing information quality of a community-based encyclopedia*. Paper presented at the Proceedings of the International Conference on Information Quality - ICIQ 2005.

Vuorikari, Riina, Manouselis, Nikos , & Duval, Erik. (2008). Using Metadata for Storing, Sharing and Reusing Evaluations for Social Recommendations: the Case of Learning Resources. *Social Information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively. Hershey, PA: Idea Group Publishing*, 87–107.

Xu, Lin, Hoos, Holger H., & Leyton-Brown, Kevin. (2007). *Hierarchical hardness models for SAT*. Paper presented at the Proceedings of the 13th international conference on Principles and practice of constraint programming, Providence, RI, USA.

Competency Comparison Relations for Recommendation in Technology Enhanced Learning Scenarios

Gilbert Paquette, Delia Rogozan, Olga Marino

CICE Research Chair
LICEF Research Center, Télé-université
gilbert.paquette@licef.ca, delia.rogozan@licef.ca,
olga.marino@licef.ca

Abstract.   In this paper, we address the problem of competency comparison, providing some heuristics to help match the competencies of users with those involved in task-based scenario components (actors, tasks, resources). Competencies are defined according to a structured competency model based on a domain ontology. We provide a context for recommendation through a learning scenario model. The approach has been implemented by extending an ontology-driven system called TELOS. It has been tested with a learning unit where these comparison relations are used to provide recommendations to users involved in a technology enhanced learning scenario.

Keywords. Adaptivity. Semantic Referencing. User Modeling. Assistance Systems. Recommendation. Personalization.

1   Introduction - The Semantic Adaptive Web

Commercially mature recommender systems have been introduced during recent years in popular e-commerce web sites such as Amazon or eBay. Yet, according to Adomavicus and Tuzhilin (2005), new developments must "include, among others, the improved modeling of users and items, and incorporation of the contextual information into the recommendation process". The new developments in Web 2.0 and the Semantic Web lead to the idea of an "Adaptive Semantic Web" (Dolog and al 2004) based on the "Web of data" (Heath and Bizer 2011; Allemang D. and Hendler J. (2011) . They open new approaches in the area of recommender systems, in particular for trust-aware recommendation, the use of folksonomies and the ontological filtering of resources (Jannach et al, 2011)

The present contribution addresses some of these issues. It proposes to provide a context for recommendation using a learning scenario model and its implementation through a structure of tasks executed by

actors using various kinds of input resources, producing outcomes and interacting with other actors (Paquette 2010). An example for Technology Enhanced Learning is presented in section 2 and used throughout the text to illustrate the main concepts involved here.

We have built an ontology-based competency model, also presented in section 2. It is used for the semantic referencing of actors, tasks and resources in a scenario, and as a basis for recommendation. Unlike other approaches for an ontology-based recommendation, such as OWL-OLM (Denaux et al. 2005) or Personal Reader (Dolog et al., 2004), this competency model extends a domain ontology with mastery levels, e.g. generic skills and performance levels.

In section 3, we describe a method for referencing resources in a learning scenario with such ontology-based competencies. We also address the central problem of competency comparison, providing some heuristics to help match a user's competencies with those possessed by other actors or involved in task or resources in a scenario.

In section 4, we present an application where these comparison relations are used to define recommendation agents, to help personalize a learning scenario. Applications like the one presented here are implemented as an extension of the TELOS ontology-driven system (Paquette and Magnan, 2008), providing a proof of concept of the general approach.

2 Competency referencing of learning scenario components.

2.1 Scenario models for learning contexts

Figure 1 presents a simple scenario model, a screen-shot from our G-MOT scenario editor (Paquette et al., 2011). There are four tasks, two actors (a professor and a student) and some resources that are input to the tasks or produced by the actor responsible (R-link) for the task. Each task is decomposed into sub-models, not shown on the figure, which describe it more precisely on one or more levels. This scenario will serve to illustrate the concepts presented in this paper.

In the first task, the student reads the general assignment for the scenario and the list of target competencies he is supposed to acquire. In the second one, he builds a table of planet properties that is validated by the professor, using the information in a PowerPoint document (called *"Planet Properties"*). In the third one, using this table assessed

by the professor *("Validated table"),* he compares five properties of planets to find out relations between properties, writing a text on his findings *("Validated relations").* In the last task he is asked to order the planets according to their distance to the Sun and to write his ideas on planets that can sustain life.

On the right side of the figure, three recommendation agents have been added to corresponding tasks, in order to provide advice and update the student's competency model with newly acquired competencies. Their action will be explained in section 4.
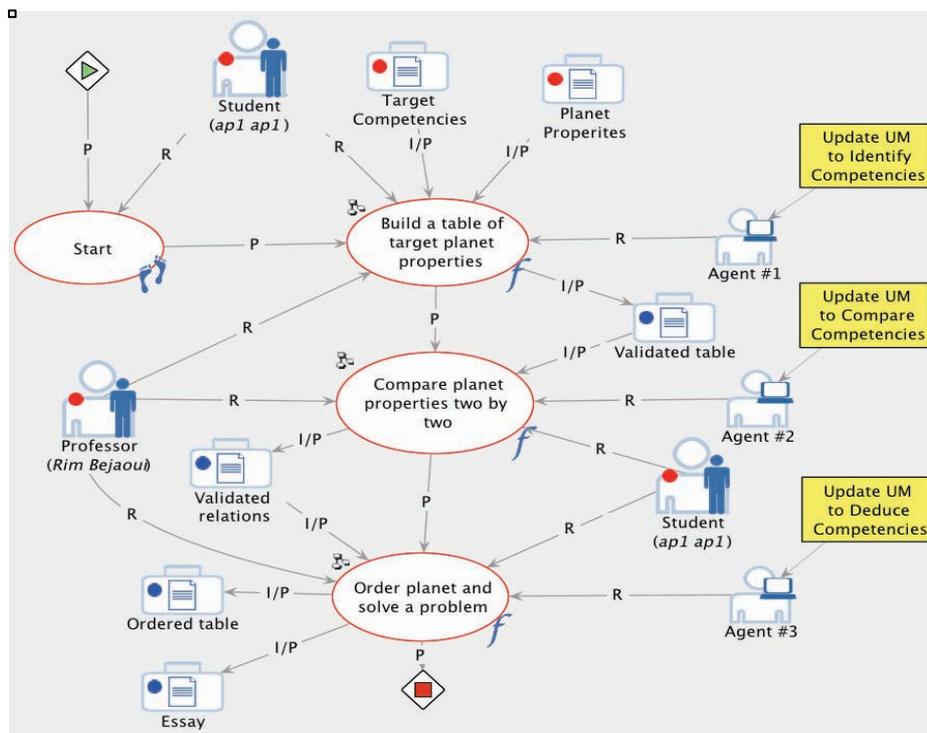
Fig. 1. An example of a scenario model

2.2 Semantic referencing of scenario components

As we have pointed out (Paquette and Marino, 2004), educational modeling languages and standards such as IMS-LD (2003) need to be improved with a structured knowledge and competency representation, in order to add semantic references to scenario components. Two main

methods are generally used: semantic references from a domain ontology or natural language statements called prerequisites and learning objectives (as in IMS-LD). Both are not sufficient for our purpose.

In most common practice, unstructured natural language statements from a competency referential are used. Such statements have many problems. First, they are not related to domain ontologies that could describe formally their knowledge part. Second, natural language statements are not appropriate for computation. Computationally, they make it difficult to reference and compare competencies assigned to actors, tasks and resources of a learning scenario. The IEEE-RCD (2007) specification allows optional definition elements as "a structured description that provides a more complete definition of the RCD than the free-form description expressed in the title and description".

Our competency model corresponds to that goal. It has been published in many conferences, journals and books, and also extensively used in instructional design projects. Devedsic (2006, p.260) describes our model as "a competency structure, corresponding to the domain ontology and represented by entry and target competencies related to the nodes in the instructional structure" (the scenario model).

Unlike other approaches based on ontologies, such as OWL-OLM (Denaux et al. 2005) or Personal Reader (Dolog et al., 2004), the proposed competency model extends a domain ontology with mastery levels, e.g. generic skills and performance levels. In fact, referencing resources with a set of concepts from a domain ontology is an important step, but generally, it is limited to "lightweight ontologies", i.e. simple taxonomies, thus ignoring the richer structures found in OWL-DL ontologies. Furthermore, to state that a person has to "know" a concept is an ambiguous statement. It does not say what exactly the person is able to do with the knowledge. It is a different competency if a user must simply recognize the malfunction of a device, diagnose it or repair it. Also, it is very different if a diagnosis is to be made in a familiar or novel situation, or with or without help.

For that purpose, in our competency model (Paquette 2007, 2010), each competency is defined as a triple (K, S, P) where K is a knowledge element (a class, a property or an individual) from a domain ontology, S is a generic skill (a verb) from a taxonomy of skills, and P is a combination of performance criteria values. This model can be instanciated to any system of competencies describing them in terms of skills, knowledge and performance, such as the European Qualification

Framework, in which qualifications range from basic level 1 to advanced level 8 (EQF 2012).

This model has been implemented (in a TELOS extension) for referencing actors, tasks and resources in the following way. The domain ontology follows the W3C OWL-DL standard. The taxonomy of skills is simplified to a 10-level scale (0-PayAttention, 1-Memorize, 2-Explicitate, 3-Transpose, 4-Apply, 5-Analyze, 6-Repair, 7-Synthetize, 8-Evaluate, 9-Self-Control). The performance part is a combination of performance criteria values with four performance levels (0.2-aware, 0,4-familiar, 0,6-productive, 0.8-expert), added to the skill level.

For example, using a domain ontology of solar system planets (shown on figure 3) and a competency referential (or model) based on this ontology, competencies can be associated to a resource from the scenario on figure 1. The competencies describing such a resource, (*"Planet Properties"*), could be compared with those of a user (*Gilbert Paquette*) to verify if he has all of them, or some, or none, in his competency model, and offer a recommendation accordingly.
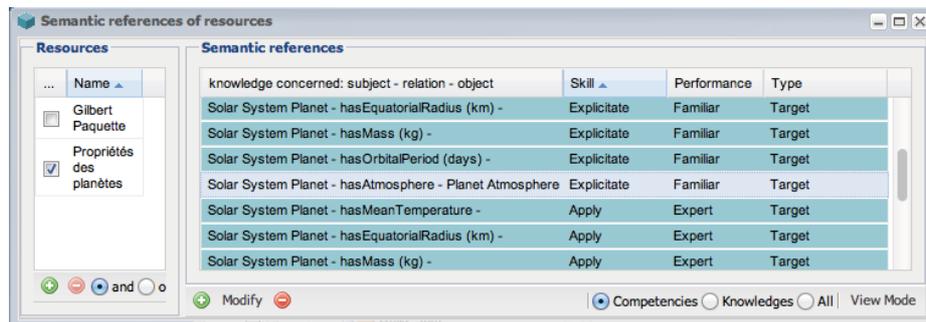
Fig. 2. An example of competency referencing for an actor and a resource

2.3 Tasks, resources and user competency models.

All components of a scenario are thus referenced using comparable competencies, based on the same domain ontology. Resources and tasks in a scenario are referenced by two sets of competencies, one for prerequisite competencies, and the other, for target competencies (i.e. learning objectives).

A *user competency model* is composed of three main parts (Moulet et al. 2008).

- The core of the model is the list of the user's actual competencies selected in one or more competency referentials. As mentioned above, each user's competencies C is described by its knowledge (K), skills (S) and performance (P) components.
- The competency model contains also documents (texts, exam results, videos, images, applications, etc.) structured into an e-portfolio that presents evidence for the achievement of related competencies.
- The context in which a competency has been achieved is also stored in the model. It includes the date of achievement, the tasks that led to it, the link to the evidence in the e-portfolio and the evaluator of this evidence.

3 Competency Comparison

3.1 Knowledge and Competency Comparison Relations.

Consider two competencies $C_1=(K_1, S_1, P_1)$ and $C_2=(K_2, S_2, P_2)$. It will be rarely the case that the three parts will coincide, but we can evaluate the semantic proximity or nearness between C_1 and C_2, based on the respective positions of their knowledge parts in the ontology and the values associated with the skills and the performance levels.

From a semantic point of view, a recommendation agent evaluates for example if a user's actual competency is very near, near, or far from the prerequisite or target competencies of a resource or a task or from the actual competencies of another user. The agent can also evaluate if a competency is stronger or weaker than another one according to the levels of its skill and performance parts. Or it can determine if the competency is more specific or more general according to the positions in the ontology of the corresponding knowledge components.

Thus, to take advantage of the competency representation, we need to establish a formal framework for the evaluation of the proximity, strength or generality of competencies. In the next section we define the semantic proximity between knowledge parts of a competency. In section 3.3 we extend the framework to competencies by considering skills and performance .

3.2 Semantic Proximity of the Knowledge Components.

In this section, we focus only on the knowledge part of two competencies to be compared. Maidel et al. (2008), proposes an approach in which a taxonomy is exploited. Five different cases of matches between a concept A in the resource profile and a concept in the user profile are considered. Various matching scores are given when a concept A in the item profile, a) is the same, b) is a parent, c) is a child, d) is a grandparent or e) is a grandchild of a concept in the user profile. Then, a similarity function is used to combine these scores in order, for example, to recommend news to a user according to his preference.

Maidel et al. state that if the use of taxonomy is not considered, the recommendation quality significantly drops. Our thesis is that, for education, taxonomy is not enough either, for only subsumption relations are exploited. We thus propose to define the semantic proximity between knowledge elements, based on their situation in the domain ontology.
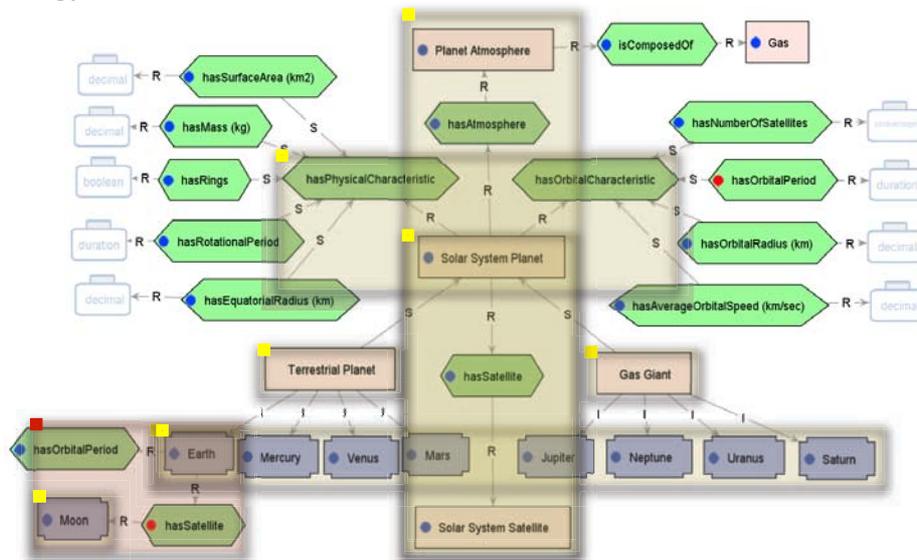
Fig. 3. Domain Ontology on Solar System Planets and some Proximity Relations

Semantic references are components from an OWL-DL ontology that describe the knowledge in a resource. A few examples of these

knowledge references are shown on figure 3 that presents part of an ontology for solar system planets[1]. They can take six different forms *solarSystemPlanet* is a class reference (C), *Neptune* is an instance reference, *solarSystemPlanet/hasAtmosphere/atmosphere* is an object property reference with its domain and range classes (D-oP-R), *Earth/hasSatellite/Moon* is an object property instance reference (I-oP-I'), *solarSystemPlanet/hasOrbitalPeriod* is a data property reference with its domain class (D-dP), *Earth/hasNumberOfSatellites* is a data property instance reference (I-dP).

We have investigated systematically these 6 forms of OWL-DL references to decide on the nearness of two references K1 and K2. For example, a concept (form C) is near its sub classes, super classes, and instances. It is also near an object or data property (forms D-oP-R and D-oP) that has a domain or range identical or equivalent to this concept. A property reference, with its domain and range (form D-oP-R) is near a sub-property or super-property with the same domain and range. It is also near to a subclass or superclass of its domain and range.

Other criteria assert when a reference K_1 is more general or more specific than another one K_2. For example, K_1 is more general than K_2 if K_1 is a superclass of K_2, has K_2 as an instance, appears as domain or range of a data or object property reference K_2, or contains an instance in the domain or range of a data or object property reference K_2.

3.3 Semantic Relationships Between Competencies.

Let us now extend the comparison between ontology components to add the skill (S) and performance (P) dimensions of the competency model. Figure 4 presents a few comparison cases between two competencies $C_1 = (K_1, S_1, P_1)$ and $C_2 = (K_2, S_2, P_2)$ in the case where K_1 is near K_2. Other cases are not considered, i.e. comparison fails.

To illustrate the heuristics, the (S, P) couples are represented on a 2-dimensional scale (figure 4). Skills are ordered from 0 to 9 and grouped into four classes as follows: $\{\{0,1\}, \{2,3,4\}, \{5,6,7\}, \{8,9\}\}$. Performance indicators are grouped into four decimal levels.

For example, a competency C1 with an analyze skill at an expert level is labeled 5.8 ($S_1 + P_1$). A competency C2 at a level 7.2 or 6.4 will

---

[1] Unlike other graphic presentation of ontologies, properties are shown as objects (hexagons) between their domain and range classes (rectangles). It this way, the relations between properties are shown on the same graph. Individual are linked to classes by an "I" link.

be considered near and stronger than C1 because the synthesize skill or the repair skill are in the same class than the analyze skill, but one or two levels higher in the skill's hierarchy. On the other hand, a competency C2 at a level 5.2 will be considered very near and weaker than C1 because it has the same skill's level but with a lower performance level. Other possible competencies in the "far zone" will be considered too far to be comparable. Also, depending on the relationship between K1 and K2, C2 will be defined as equivalent, more general or more specific than C1. These relations between competencies can also be combined to define more complex relationships. For instance, it is possible for a competency reference to be near and stronger and more general than another one.
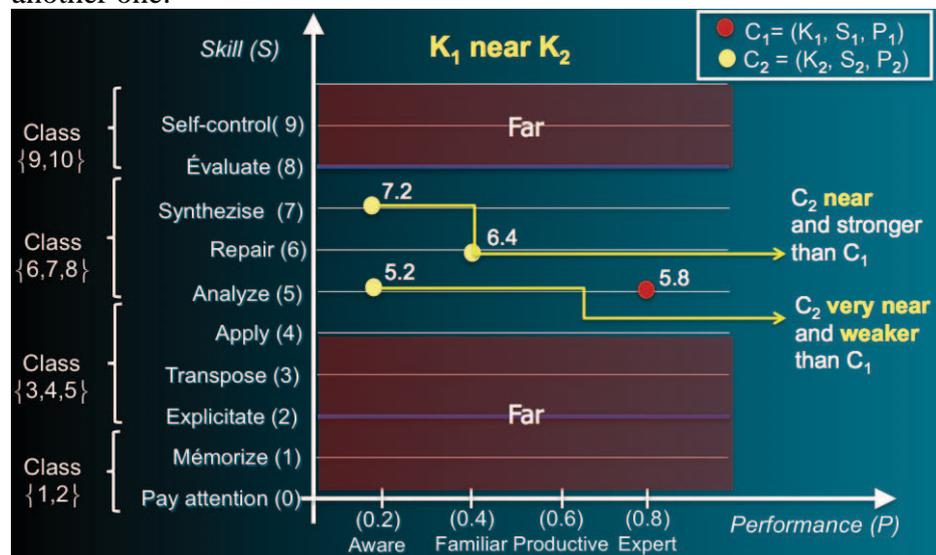
Fig. 4. Comparison criteria for two competencies with their knowledge parts near.

4   Recommendation based on competency comparison

4.1   Competency-based conditions and rules.

Recommendation agents are added to a scenario, linked to some of the tasks called *insertion points,* as in the example of figure 1. The designer defines these agents by a set of rules. In each rule, one and only one of the actors linked to the task at the insertion point is chosen as the

receiver of the recommendation. If a triggering event occurs at run time such as "task completed", "resource opened", etc., each applicable rule condition is evaluated and its actions are triggered or not, depending on the evaluation of the condition.

A competency-based condition takes the form of a triple:

- *Quantification* takes two values: HasOne or HasAll, which are abbreviations for "the user has one (or all) of its competencies in some *relation* with an object competency list".
- *Relation* is one of the comparison relations between semantic references presented above: Identical, Near/Generic, Near/Specific, VeryNear/Generic, VeryNear/Specific, Stronger, Weaker; or any combination of these relations.
- *ObjectCompetencyList* is the list of prerequisite or target competencies of a task or a resource at/around the insertion point.

Lets take the example of a condition like:

HasAll /NearMoreSpecific / Target competencies for Essay

When it is evaluated, competencies in the user's model are retrieved, together with the list of target competencies for the resource "Essay" The evaluation of the relation "NearMoreSpecific" provides a true or false value according to the method exposed in section 3.3.

4.2 Recommendation actions, an application.

The action part of an agent's rule can perform one or more tasks: give advice to the actor, notify another actor, recommend various learning resources, update the user's model, propose to jump to another task or to another learning scenario.

All these possibilities have been implemented. On figure 1, we have presented a scenario with three recommendation agents. For example, Recommender agent #1 on figure 1 will verify if the student has succeeded the second task in the scenario ("Build a table…"). It has 3 rules, shown on the screen-shot of figure 5.

The rule "Update User Model" transfers the list of target competencies for the task to the student's user model if he has succeeded to build a validated table of planet properties. If he has failed, a second rule will send a notification to the professor to interact with the student. Finally, a third rule provides an advice to the student and recommend consulting a resource shown on figure 5.
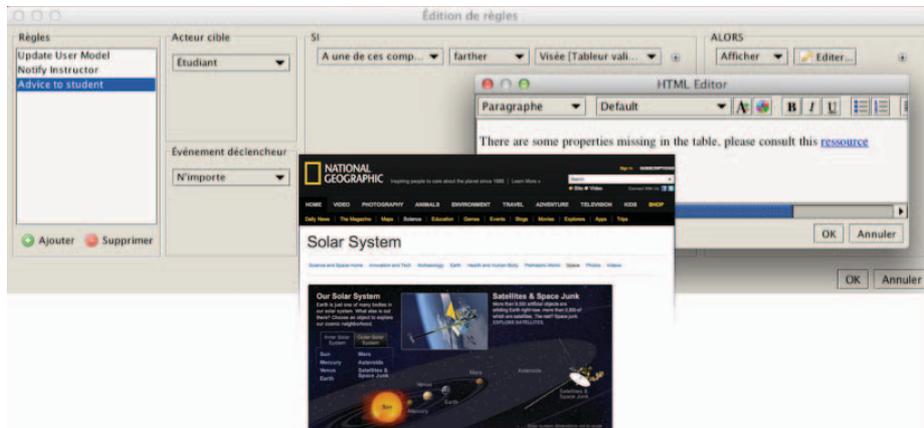
Fig. 5. Example of an agent's rule based for updating a user's competency model

5 Conclusion

We have produced an implementation for competency-based assistance that has been tested with a few scenarios and recommendation situations. It provides a proof of concept for the general method. It also provides a workbench to investigate further and extend the methods presented here with variants and a larger range of applications.

First of all, extensive experimental validation will help refine the relation for semantic nearness between OWL-DL references. Adding weights to the various cases would improve the quality of the evaluations. For example, one could assert that a subclass or superclass is closer to a class than its instances or one of its defining properties, especially if there are many defining attributes for this class.

Our model of multi-actor learning scenarios embeds the idea of collaboration between learners, and between learners and various kinds of facilitators. Recommendation for groups in collaborative scenarios has not been thoroughly explored yet.

Finally, to improve the practical use of approach, some of the task will have to be partly automated and the ergonomics of the system improved. Still, the approach presented here sets the ground for an open and flexible method for semantically aware recommendation systems.

6   References

1. Adomavicus, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6).
2. Allemang D. and Hendler J. (2011) *Semantic Web for the Working Ontologist* (2nd Edition), Morgan-Kaufmann, Elevier Inc.
3. Denaux, R., Aroyo, L. and Dimitrova, V. (2005) *An approach for Ontology-based Elicitation of Usr Models to Enable Personalization on the Semantic Web*, Retrieved June 2012, from http://wwwconference.org/www2005/cdrom/docs/p1170.pdf
4. Devedzic, V. (2006) *Semantic Web and Education*. Springer's Integrated Series in Information Systems, 353 pages.
5. Dolog, P., Henze, N., Nejdl, W., & Sintek, M. (2004). *Towards the adaptive Semantic Web.* Lecture Notes in Computer Science. Berlin, Germany: Springer.
6. Dolog, P., Henze, N., Nejdl, W., & Sintek, M. (2004). The Personal Reader: Personalizing and Enriching Learning Resources using Semantic Web Technologies, *AH 2004: International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, 2004. Retrieved June 2012 from:
   http://www.kbs.unihannover.de/Arbeiten/Publikationen/2004/ah2004_d_h_n_s.pdf
7. IEEE-RCD (2007). *LTSO Contents – IEE RCD Overview*. http://www.cen-ltso.net/main.aspx?put=264
8. *The European Qualification Framework*. http://ec.europa.eu/education/lifelong-learning-policy/eqf_en.htm. Retrieved August 29, 2012
9. Heath T. and Bizer C. (2011) *Linked Data – Evolving the Web into a Global Data Space.* Synthesis Lectures on The Semantic Web: Theory and Technology, Morgan and Claypool Publishers,
10. Jannack, D., Zanker, M., Felfering A., & Friedrich G. (2011) *Recommender Systems, an Introduction.* Cambridge University Press. New York, USA, 335 pages.
11. Maidel R., Shoval P., Shapira B., & Taieb-Maiomon M. (2008) Evaluation of an Ontology-content based filtering method for a personalized newspaper, *Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys '08),* Lausanne, Switzerland, pp. 91-98
12. Moulet, L., Marino, O., Hotte, R., & Labat, J.- M. (2008). Framework for a competency-driven, multi-viewpoint and evolving learner model. In B. Woolf, et al. (Eds.), *Proceedings ITS 2008, 9th International Conference Intelligent Tutoring Systems,* (LNCS 5091, pp. 702-705). Montréal, Canada: Springer.
13. Paquette, G., Léonard, M. Lundgren-Cayrol, K. (2011) The MOT+ Visual Language for Knowledge-Based Instructional design, in *Instructional Design: Concepts, Methodologies, Tools and Application*. Information Resources Management Association (IRMA), IGI-Global, 2011, pp 697-717
14. Paquette, G. 2010. *Visual Knowledge Modeling for Semantic Web Technology*. IGI Global, 433 pages
15. Paquette, G., & Magnan, F. (2008). An executable model for virtual campus environments. In H. H. Adelsberger, Kinshuk, J. M. Pawlowski, & D. Sampson (Eds.), *International handbook on Information Technologies for education and training, 2nd edition*, (pp. 365-405). Springer
16. Paquette G. and O. Marino (2006), "Learning Objects, Collaborative Learning Designs and Knowledge Representation", in *Technology, Instruction., Cognition and Learning* , Vol. 3, p.85-108, Old City Publishing, Inc.

Revisiting the Multi-Criteria Recommender System of a Learning Portal

Nikos Manouselis[1], Giorgos Kyrgiazos[2], Giannis Stoitsis[1]

[1]Agro-Know Technologies
17 Grammou Str., 152 35, Athens, Greece
{nikosm, stoitsis}@ieee.org

[2]Computer Technology Institute and Press - Diophantus
10 Davaki Srt., 115 26, Athens, Greece
kyrgiazos@cti.gr

Abstract. Results of previous studies have indicated that the same recommendation algorithms perform in totally different ways when a different dataset is considered, thus leading to the need for continuous monitoring of how algorithms perform in a realistic and evolving setting. In this paper we investigate such a real life implementation of a multi-criteria recommender system and try to identify adjustments that need to take place in order for it to better match the requirements of its operational environment. More specifically, we examine the case of a multi-attribute collaborative filtering algorithm that has been supporting the recommendation service within a Web portal for organic and sustainable education.

Keywords: Multi-criteria recommender system, experimental investigation, real life deployment, learning portal

1 Introduction

In the domain of education, recommender systems have been introduced more than a decade ago, with deployed and well-studied systems like Altered Vista (Recker et al., 2003) and CoFIND (Dron et al., 2000). Additionally, surveys of the systems that have been actually implemented in a real life setting indicate that they are very few (Manouselis et al., 2011; Manouselis et al., 2012; Verbert et al., 2012). For instance, a very recent analysis of existing educational recommenders that has been carried out by Manouselis et al. (2012) revealed that out of the forty two (42) systems proposed in the literature since 2000, only thirteen (13) have been actually deployed as a fully implemented and operational system – and not all in a real usage setting. This significantly inhibits the quality of research that can take place in this application domain, since it is important to be able to study the social and psychological requirements on how people react to and act upon recommender systems for the learning sciences (Buder & Schwind, 2012). It also implies that the deployment of a

real world recommender system in education is a demanding and challenging exercise.

In this paper, we try to reflect on one of the main questions that the people responsible for an operational recommender system need to face: how can we monitor, test, and fine-tune the algorithms deployed in a real setting, by using data from its actual operation. More specifically, we focus on the case of an existing educational recommender system that collects data that educators and learners provide on digital content that may be used to support education and research on organic and sustainable agriculture, and uses this dataset to provide recommendations about relevant resources (Manouselis et al., 2009). Our study particularly focuses on the collaborative filtering algorithm that has been chosen and parameterized to collect multi-criteria ratings on the content items in order to recommend new items to the users, and tries to investigate two dimensions:

- How does the implemented algorithm perform over a current rating data set from the targeted educational application, also compared to some alternative multi-criteria recommendation algorithms;
- How do the studied algorithms perform over a synthetic data set that simulates how the users of the targeted application will have rated the available content items in a future time instance.

The remainder of this paper is structured as it follows. Section 2 provides the background of this study as it introduces collaborative filtering using multi-criteria ratings, as well as the specific educational application that serves as a case study. Then, Section 3 presents the methodology of this study, by describing the experimental setting and environment in which the study took place, the multi-criteria recommendation algorithms compared, the metrics used for their comparison, as well as the multi-criteria rating data sets that have served as the comparison basis. Section 4 presents the results of our experimental investigation, particularly presenting how the studied algorithms performed over each data set. A discussion of the results, their implication on the implemented real world service, as well as the limitations of this study is included in Section 5. Finally, some overall conclusions and directions of future research are given.

2 Background

2.1 Multi-Criteria Collaborative Filtering

In most recommender systems, the utility function usually considers a single-criterion value, e.g., an overall evaluation or rating of an item by a user. In recent work, this assumption has been considered as limited (Adomavicius & Kwon, 2007; Adomavicius et al., 2011), because the suitability of the recommended item for a particular user may depend on more than one utility-related aspect that the user takes into consideration when making the choice. Particularly in systems where recommendations are based on the opinion of others, the incorporation of multiple

criteria that can affect the users' opinions may lead to more accurate recommendations. Thus, the additional information provided by multiple dimensions or criteria could help to improve the quality of recommendations because it makes it possible to represent more complex preferences of each user. Recommender systems have already adopted multiple criteria as relevant research indicates (Adomavicius et al., 2011; Lakiotaki et al., 2011). A recent survey by Adomavicius et al. (2011) identified more than fifty (50) such systems that can be broadly classified as multi-criteria recommender ones.

Multi-criteria collaborative filtering is an extension of traditional collaborative filtering systems that is based on ratings expressed over multiple dimensions describing an item (Adomavicius et al., 2011). They allow a user to specify his individual preferences by rating each item upon multiple criteria, and then recommend to the user the items that can best reflect the user's individual preferences based on the multi-criteria ratings provided by this and other users. In single-attribute (or single-criterion) collaborative filtering, the problem space can be formulated as a matrix of users versus items (or user-rating matrix), with each cell storing a user's rating on a specific item. The recommender estimates a utility function R for the entire or some part of the Users   Items space based on known ratings and possibly other information (such as user profiles and/or item features). Collaborative filtering aims to predict this utility R of items for a particular user (called active user) based on the items previously evaluated by other users. That is, the utility $R(a,i)$ of item i for the active user $a \in Users$ is estimated based on the utilities $R(u,i)$ assigned to item i by those users $u \in Users$ who are "similar" to user a.

The difference to single-criterion rating systems is that the utility function $R(u,i)$ is the total utility of an item, calculated by synthesizing the partial utilities of the item on each one of the rating dimensions (or criteria). Assuming that there is no uncertainty during the decision process, the total utility of an item $i \in Items$ for a user $u \in Users$ is often expressed as an additive value function of the evaluation or ratings $g(u,i)$ that user u provides for item i over each one of the k criteria, such as:

$$R(u,i) = \sum_{1}^{k} g(u,i) \qquad (1)$$

Such a linear form of the total utility function is the simplest and most popular form of a utility function. Other forms that could be used include an ideal point model, dependencies and correlations, as well as diminishing utility forms (Price & Messinger, 2005).

The collaborative filtering techniques that use multi-criteria ratings to predict an overall rating and/or individual criteria ratings can be classified by the formation of the utility function into two categories: heuristic-based (sometimes also referred to as memory-based) and model-based techniques (Adomavicius et al., 2011). Heuristic-based techniques compute the utility of each item for a user on the fly based on the observed data of the user and are typically based on a certain heuristic assumption. In contrast, model-based techniques learn a predictive model, typically using statistical or machine-learning methods that can best explain the observed data, and then use the learned model to estimate the utility of unknown items for recommendations.

Different approaches may be also followed by the algorithms developed to support multi-criteria collaborative filtering. For instance, algorithms may (Adomavicius & Kwon, 2007; Manouselis & Costopoulou, 2007):

- try to predict the total utility for an item using the total utility values that the item has for other users;
- try to calculate a separate prediction per each criterion and then use the utility function in order to acquire the predicted total utility.

2.2 Case Study

In this paper, we focus on the particular case of a real life implementation of a multi-criteria recommender system in the context of an educational application. This is the case of the Organic.Edunet Web portal for agricultural and sustainable education (http://www.organic-edunet.eu) that was launched in 2010. Its aim has been to facilitate access, usage and exploitation of digital educational content related to Organic Agriculture (OA) and Agroecology (AE). In order to achieve this aim, it networked existing collections with educational content on relevant topics from various content providers, into a large federation where content resources are described according to standard-complying metadata.

After more than two (2) years of operation, Organic.Edunet seems to be established as a reference source of educators and researchers working on relevant topics. It has already attracted more than 20,000 unique visitors from about 150 countries. About 5,000 visitors have registered into the portal's community, being able to receive regular information updates related to the portal and its content, as well as having access to personalized services such as receiving recommendations about potentially interesting content resources. The recommendation service in Organic.Edunet is supported by two separate algorithms that are using different data as input and are currently running independently (Manouselis et al., 2009): a content-based recommender using tags and textual reviews as input; and a multi-criteria collaborative filtering system that uses as input the ratings that users provide over three criteria: Subject Relevance, Educational Value and Metadata Quality.

This study focuses on the multi-criteria algorithm and the recommendations that it produces. This algorithm was proposed by Manouselis & Costopoulou (2007) as a multi-criteria extension to typical heuristic neighborhood-based algorithms that may be found in the collaborative filtering literature. It follows the generic steps of Herlocker et al. (2002) in order to calculate a prediction:

- *Stage A - Similarity Calculation:* similarity between the examined user (active user) and the rest of the users is calculated using some similarity measure;
- *Stage B - Feature Weighting:* further weight similarity according to the characteristics of each examined user or some heuristic rules;
- *Stage C - Neighborhood Formation/Selection:* select the set of users to be considered for producing the prediction;

- *Stage D - Combining Ratings for Prediction:* normalize the ratings that the users in the neighborhood have provided for the unknown item, and use some method to combine them in order to predict its utility for the active user.

The implemented multi-criteria extension is called the **Similarity per evaluation (PG)** algorithm. It calculates the prediction of the total utility $R(a,i)$ of a target item $i \in Items$, by calculating k predictions of how the active user would evaluate i upon each one of the criteria, and then synthesizes these predictions into a total utility value.

Since the implementation of the recommendation algorithm took place during the design stage of the portal, we based our selection on the experience from a lab testing experiment that took place using an existing data set from another learning portal (Manouselis et al., 2010). Results of the simulated execution of more than 360 variations of the PG algorithm over this data set indicated that it would make sense to implement a version that: uses a Cosine/Vector distance function to measure similarity between users; engages a Correlation Weight Threshold (CWT) to select users that have similarity value of more than 0.5 for the neighborhood; and calculates predicted ratings as a weighted mean of the ratings that the neighbors have given over an unknown item. This variation has shown to achieve a Mean Absolute Error (MAE) over the prediction of less than 0.7 (in a scale 1-5) and a coverage close to 70% of the items.

Nevertheless, the fact that the specific algorithm or variation performed well over a data set coming from a similar application context (that is, of a portal with learning resources) does not mean that it would also perform well during the operation of the Organic.Edunet portal. There are several reasons for this:

- The properties of the users vs. items matrix of Organic.Edunet may be different than the ones of the dataset of the other application.
- The properties of the Organic.Edunet matrix may change/evolve with time.
- Alternative algorithms (e.g. new ones proposed in literature) that were not included in the initial experimentation may prove to perform better than the one selected.

To this end, we decided to repeat the experimental investigation of candidate algorithms for the Organic.Edunet portal, using additional algorithms as options, as well as a synthetic data set that tries to mimic a future state where users will have provided more multi-criteria ratings over the educational resources.

3 Methodology

3.1 Experimental Setting

The main goal of the experimental testing has been to investigate the performance of different variations of both the algorithm currently implemented in Organic.Edunet as well as alternative multi-criteria recommendation algorithms. The specific objectives have been:

- To use a current instance of the users vs. items matrix of Organic.Edunet in order to execute all candidate variations and measure their expected performance.
- To generate a synthetic data set that mimics an instance of the Organic.Edunet community in the future, and explore if performance of the candidate algorithms would be expected to change in the future.

The evaluation protocol follows the typical steps of offline experiments with pre-collected or simulated data that Shani & Gunawardana (2011) also described for testing the performance of candidate algorithms. Generally speaking, our experiment follows the approach of similar experiments in other domains (Herlocker et al., 2004) or education (Lemire et al., 2005; Sicilia et al., 2010). The following paragraphs describe the settings, methods and tools of the experimental investigation.

   The offline experiment took place using a software environment that has been specifically developed and used for the simulation of multi-criteria recommender systems, called the *Collaborative Filtering Simulator (CollaFiS).* This environment allows for importing various data sets, parameterizing candidate algorithms, executing them and measuring expected performance using multiple performance metrics (Manouselis & Costopoulou, 2006). The *CollaFiS* environment has been extended to support the algorithms and metrics that are particularly studied in this experiment, as described later. *CollaFiS* provides the option for experimentally testing the multi-criteria algorithms proposed by Manouselis & Costopoulou (2007). We have extended the previous implementation of *CollaFiS* in order to also include some algorithms proposed by Adomavicius & Kwon (2007). Overall, the studied algorithms included:

- the *Similarity per evaluation (PG)* algorithm (currently implemented in Organic.Edunet) that calculates similarity separately upon each criterion, predicts the rating also separately upon each criterion, and then is synthesizing the predictions into a total predicted utility;
- the *Average Similarity (AS) and Minimum or Worst-case Similarity (WS)* algorithm versions proposed by Adomavicius & Kwon (2007) that use either the average or the minimum of the similarities over each criterion in order to calculate the total predicted utility;
- some *Non-personalised* algorithms as a basic comparison, such as giving random values as predictions or calculating an arithmetic, geometrical or deviation-from-mean weighted sum of all past evaluations.

For the personalized algorithms *(PG, AS, WS)* we have considered the following design options in order to study different variations within the generic steps of Herlocker et al. (2002) described in section 2.2:

- *during Stage A - Similarity Calculation:* examined the calculation of the similarity using the Euclidian, Vector/Cosine, and Pearson distance functions as options.
- *during Stage C - Neighborhood Formation/Selection:* examined both the use of a *Correlation Weight Threshold* (CWT) for the similarity value as a selector of potential neighbors, as well as of an absolute value for the *Maximum Number of Neighbors* (MNN).
- *during Stage D - Combining Ratings for Prediction:* examined three different options for synthesizing partial utilities, i.e. calculating the prediction as a simple arithmetic mean, as a mean weighted by the similarity value, as well as a normalized weighted mean that takes into consideration also the deviation from the arithmetic mean (as Herlocker et al. 2002 suggest).

This led to 18 variations of each examined algorithm. By also experimenting with various values for the CWT (20 variations between '0' and '1' as a threshold) and MNN (20 variations using '1' to '20' maximum neighbors) parameters, the number grew to more than 1,080 algorithmic variations explored in total.

There are several performance metrics used in the literature. In this experiment we examined the following evaluation metrics that *CollaFiS* incorporates:

- *Accuracy*: to measure the predictive accuracy of the multi-criteria algorithms, we calculated the mean-absolute error (MAE). MAE is the most frequently used metric when evaluating recommender systems. Herlocker et al. (2004) have demonstrated that since it is strongly correlated with many other proposed metrics for recommender systems, it can be preferred as easier to measure, having also well understood significance measures.
- *Coverage*: to measure the coverage of the multi-criteria algorithms, we calculated the items for which an algorithm could produce a recommendation, as a percentage of the total number of items. Herlocker et al. (2004) recommend the measurement of coverage in combination with accuracy.

Two different data sets have been used to support the simulated execution of the algorithms. Both have been imported into *CollaFiS* and appropriately processed. To facilitate the execution of the experiments, they have been split into one training and one testing component (using an 80%–20% split).

The first data set (OEreal) has been a recent export/instance of the users vs. items matrix of Organic.Edunet, with the collected multi-criteria ratings that the users of the portal have provided over the content items. As mentioned before, Organic.Edunet collects user evaluations over three criteria that are all rated using a discrete scale from 1 to 5. In this real dataset, 99 users have provided 477 multi-criteria ratings over 345 items.

The second data set was a simulated one (OEsim) that tried to represent a future state of the *Users x Items* matrix of Organic.Edunet. More specifically, the

distributions of the ratings of the OAreal dataset were taken as input to a Monte Carlo generator of random multi-criteria ratings of the same users. The considered scenario is that the current users that have been rating a sample of the Organic.Edunet items provide more ratings on this specific sample of already rated items in order to make it more dense. The produced synthetic dataset incorporates the original real one, has the same number of users and items, and includes a total number of 1,280 multi-criteria ratings. In a similar way, alternative scenarios could be considered, with more users and/or items, and with more dense or sparse data sets.

Table 1. Non-personalized algorithms over OEreal and OEsim data sets

Variation	Pure Random	Random Existing Rating	Arithmetic Mean	Geometric Mean	Deviation-from-Mean
MAE over OEreal	1.59	1.33	1.28	1.27	1.03
Coverage over OEreal	100%	100%	37.89%	37.89%	32.63%
MAE over OEsim	1.38	1.01	0.86	0.89	0.83
Coverage over OEsim	100%	100%	96.09%	96.09%	96.09%

Table 2. Top-5 CWT and top-5 MNN variations over OEreal and OEsim data sets

Algorithm	Similarity	Normalization method	AVG Coverage	AVG MAE
Top-5 MNN variations over OEreal dataset				
PG	Cosine	Deviation-from-Mean	18.95%	0.9928
PG	Euclidian	Simple Mean	18.95%	1.3194
PG	Cosine	Weighted Mean	18.95%	1.3337
AS	Euclidian	Deviation-from-Mean	18.95%	1.5008
WS	Cosine	Deviation-from-Mean	18.95%	1.6886
Top-5 CWT variations over OEreal dataset				
PG	Cosine	Deviation-from-Mean	16.32%	0.8650
PG	Cosine	Simple Mean	16.32%	1.1831
AS	Cosine	Deviation-from-Mean	16.95%	1.5202
WS	Cosine	Deviation-from-Mean	16.37%	1.7316
AS	Cosine	Simple Mean	16.95%	2.1074

Top-5 MNN variations over OEsim dataset				
PG	Euclidian	Simple Mean	0.6133	0.8626
PG	Cosine	Simple Mean	0.6133	0.8653
PG	Cosine	Deviation-from-Mean	0.6133	0.8855
AS	Euclidian	Deviation-from-Mean	0.6133	1.2972
WS	Euclidian	Deviation-from-Mean	0.6133	1.8486
Top-5 CWT variations over OEsim dataset				
PG	Cosine	Simple Mean	0.5791	0.8673
PG	Cosine	Weighted Mean	0.5791	0.8681
PG	Cosine	Deviation-from-Mean	0.5791	0.8908
AS	Cosine	Deviation-from-Mean	0.5934	1.2983
AS	Cosine	Weighted Mean	0.5934	2.2086

4 Results

In this section we will present the results that have been produced by the CollaFiS tool, after executing all the studied variations of the algorithms. For each dataset we are going to present top-5 sets of options, based on metrics mentioned above.

4.1 Real Data Set

The execution of the candidate algorithms over the OEreal dataset did not provide very good results. It seems that the majority of the tested variations performed a bit better than the non-personalised algorithms (Table 1) but still with a very low coverage that was in the vicinity of 16%-18% of the items for which a prediction needed to be made. As Table 2 shows, there are some variations (like **the PG Cosine Deviation-from-Mean** with both MNN and CWT parameters) that had an acceptable MAE that is below '1'. Still we consider this error to be rather high for an operational recommender system. These results imply that the performance of any algorithm would be judged not satisfactory if only the current data set of Organic.Edunet was used for experimentation.

In Figures 1 and 2 (and in all diagrams of this paper), the continuous lines are used to illustrate the performance of *PG* variations, the heavy dashed ones the *AS* variations, and the lightly dashed ones the *WS* variations. These diagrams illustrate that the *PG* variations seem to be generally performing better than the *AS* and *WS* ones. Nevertheless, this performance seems to be rather low over the OEreal data set.
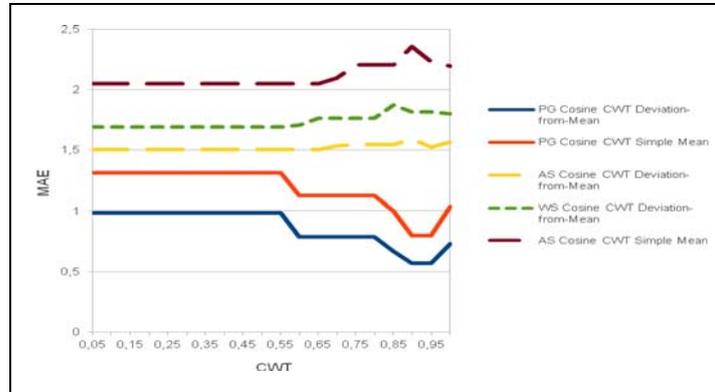
Fig. 1. MAE metric performance for top-5 CWT variations over OEreal data set
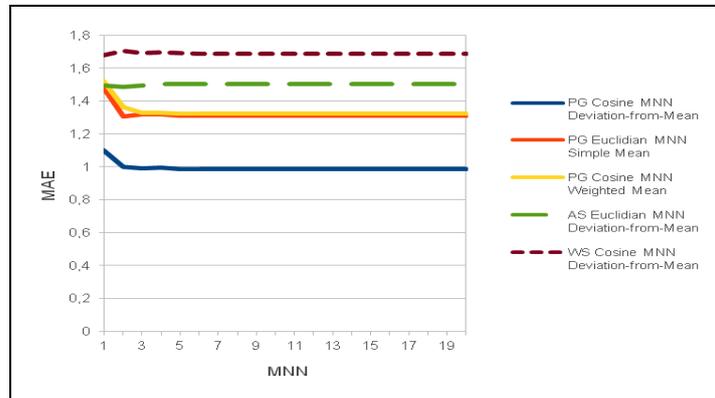
Fig. 2. MAE metric performance values for top-5 MNN variations over OEreal data set

4.2 Synthetic Data Set

The execution of the candidate algorithms over the synthetic OEsim dataset seemed to perform much better over the original OEreal one, as one would have expected (since a more dense version of the dataset has been created). As it is also shown in Table 2, the majority of the outstanding variations have a rather good coverage that is close to (for CWT) and more than (for MNN) 60%.

Surprisingly the MAE results seem to be at the level of the *non-personalised* algorithms (also presented in Table 1) and around 0.86 for the *PG MNN Euclidian Simple Mean* and the *PG CWT Cosine Simple Mean*. It seems that very simple algorithms that create weighted sums of the past ratings, such as the *Arithmetic Mean* and the *Geometrical Mean*, may provide predictions that have less MAE than the collaborative filtering variations. This could be due to the fact that the simulated users have provided additional ratings with similar distributions but still such a simplistic interpretation of this observation is not enough. Again, the graphical illustrations of

Figures 3 and 4 show that in most cases the *PG* variations seem to be performing better than the *AS* and *WS* ones, although the differences are small. To further investigate which would be the more appropriate algorithm variations to support recommendation in Organic.Edunet in such a future scenario, we did an additional experimental analysis.
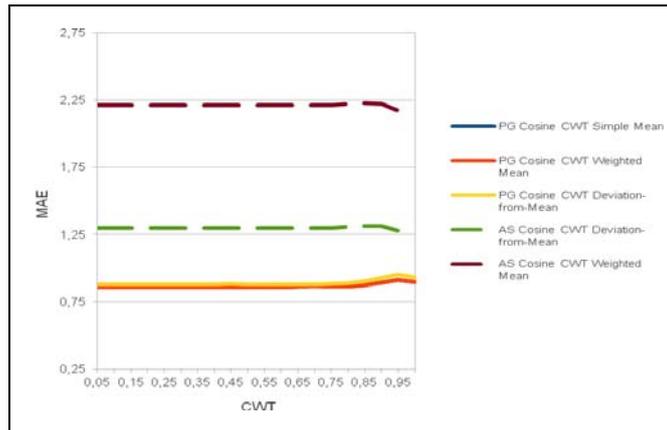
Fig. 3. MAE metric performance for top-5 CWT variations from all candidate variations over OEsim data set
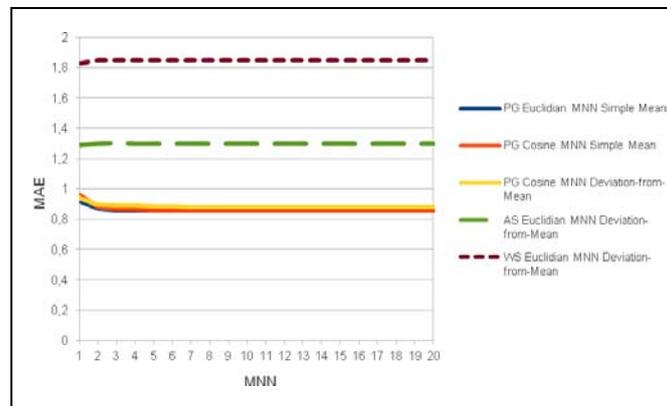
Fig. 4. MAE metric performance for top-5 MNN variations from all candidate variations over OEsim data set

More specifically, we investigated the performance of the two algorithms that performed better over the real dataset (OEreal) also over the synthetic OEsim. In this way we tried to examine if some of the algorithm variations that performed in a good way over current data, would also perform in a similar way over a future state of Organic.Edunet. As illustrated in Table 3, these algorithm variations seemed to also perform in a satisfactory way over the OEsim data. Some of them seem to be common across all data sets, with most prominent being the *PG Cosine Deviation-from-Mean* variation.

Table 3. Performance of top-2 CWT and top-2 MNN variations of OEreal data set over the OEsim one

Algorithm	Similarity	Normalization method	AVG Coverage	AVG MAE
MNN variations				
PG	Cosine	Deviation-from-Mean	61.33%	0.8855
PG	Euclidian	Simple Mean	61.33%	0.8626
CWT variations				
PG	Cosine	Deviation-from-Mean	57.91%	0.8908
PG	Cosine	Simple Mean	57.91%	0.8673

The experimental analysis overall indicates that the PG algorithm currently implemented in the Organic.Edunet portal is still a good choice. Yet, its exact parameterization and fine-tuning so that the right values are chosen that will give better results, is an exercise that needs to be taking place quite often in such a changing environment. As the community of users grows, the properties of the Users x Items matrix (that is, of the dataset) will be dynamically changing. For instance, during the past year only, more than 1,000 new users have registered in the portal. In addition, the content collections to which the portal gives access to, is ready to expand from about 11,000 items to some 30,000.

This calls for careful consideration and planning from the perspective of the designer and operator of the recommendation service. One option would be to run frequent offline experiments with most recent updates of the data set, in order to find which algorithm variations is more appropriate every time for the application. Another approach would be the investigation of adaptive algorithms that will automatically measure their performance (e.g. the accuracy and coverage of their predictions) over a dataset with specific properties, and then adapt their parameters in order to achieve better results. Such an approach can be a rather computationally-demanding task that calls for a re-engineering of the existing recommendation service of the portal and maybe an investigation of new multi-criteria recommendation algorithms.

5. Conclusions

In this paper we investigated how the recommendation algorithm used in a real life implementation of a multi-criteria recommender system performs under various experimental conditions, by using as input different datasets with multi-criteria ratings. The case study has been a portal for organic and sustainable education, and the experimentation tested a wide number of variations with one real dataset and one synthetic one. The results indicated that some particular variations seem to perform in a satisfactory way over both datasets. This was an interesting observation, considering that in related work we have witnessed significant alterations in the performance of the same algorithms over different datasets (Manouselis & Costopoulou 2007;

Manouselis et al., 2010). It gave useful input regarding the improvements that need to be made in the algorithm currently implemented in Organic.Edunet.

Our future work includes a more extensive experiment where the correlation between the various algorithmic parameters and options and the properties of the data sets will be explored. The currently available real data sets will be used as generators of a large number of synthetic data sets with varying properties. Then the *CollaFiS* simulator will be used to execute a large number of variations and measure how they perform over the various data sets. Additional metrics will also be engaged, such as typical ones found in information retrieval (e.g. Precision, Recall, F-measure).

6. Acknowledgements

References

1. Adomavicius, G., Kwon, Y.: New Recommendation Techniques for Multi-Criteria Rating Systems. In: IEEE Intelligent Systems, 22(3), 48-55 (May/June 2007) .
2. Adomavicius, G., Manouselis, N., Kwon Y.: Multi-Criteria Recommender Systems. In Kantor, P., Ricci, F., Rokach, L., Shapira, B. (Eds.), Recommender Systems Handbook: A Complete Guide for Research Scientists & Practitioners, Springer, 769-803 (2011)
3. Buder, J., Schwind, C.: Learning with personalized recommender systems: A psychological view. In: Computers in Human Behavior, 28, 207-216 (2012)
4. Dron, J., Mitchell, R., Siviter, P., Boyne, C.: CoFIND-an experiment in n-dimensional collaborative filtering. In: Journal of Network and Computer Applications, 23(2), 131-142 (2000)
5. Herlocker, J., Konstan, J., Riedl, J.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. In: Information Retrieval, 5, 287–310 (2002)
6. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating Collaborative Filtering Recommender Systems. In: ACM Transactions on Information Systems, 22(1), 553 (2004)
7. Lakiotaki, K., Matsatsinis, N.F., Tsoukiàs, A.: Multi-Criteria User Modeling in Recommender Systems. In: IEEE Intelligent Systems, 26(2), 64-76 (2011)
8. Lemire, D., Boley, H., McGrath, S., Ball, M.: Collaborative Filtering and Inference Rules for Context-Aware Learning Object Recommendation. In: International Journal of Interactive Technology and Smart Education, 2(3) (2005)
9. Manouselis, N., Costopoulou, C.: Designing a Web-based Testing Tool for Multi-Criteria Recommender Systems. In: Engineering Letters, Special Issue on Web Engineering, 13(3), http://www.engineeringletters.com/issues_v13/issue_3/EL_13_3_22.pdf (2006)

10. Manouselis, N., Costopoulou, C.: Experimental Analysis of Design Choices in Multi-Attribute Utility Collaborative Filtering. In: International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), 21(2), 311-331 (2007)

11. Manouselis, N., Drachsler, H., Verbert, K., Duval, E.: Recommender Systems for Learning. In: Springer Briefs (2012)

12. Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., Koper, R.: Recommender Systems in Technology Enhanced Learning. In: Kantor, P, Ricci, F, Rokach, L, Shapira, B (eds), Recommender Systems Handbook, pp. 387-415, Springer US (2011)

13. Manouselis, N., Kosmopoulos, T., Kastrantas, K.: Developing a Recommendation Web Service for a Federation of Learning Repositories. In: Proc. of the International Conference on Intelligent Networking and Collaborative Systems (INCoS), IEEE Press (2009)

14. Manouselis, N., Vuorikari, R., Van Assche, F.: Collaborative Recommendation of e-Learning Resources: An Experimental Investigation. In: Journal of Computer Assisted Learning, 26(4), 227-242 (2010)

15. Price, B, Messinger, P.R.: Optimal Recommendation Sets: Covering Uncertainity over User Preferences. In: Informs Annual Meeting: Denver 2004, October 24-27 2004, AAAI Press (2005)

16. Recker, M.M., Walker, A., Lawless, K.: What do you recommend? Implementation and analyses of collaborative information filtering of web resources for education. In: Instructional Science, 31(4/5), 299–316 (2003)

17. Shani, G., Gunawardana, A. Evaluating Recommendation Systems. In Kantor, P, Ricci, F, Rokach, L, Shapira, B (eds), Recommender Systems Handbook, 257-297, Springer US, (2011)

18. Sicilia, M.-A., Garca-Barriocanal, E., Sanchez-Alonso, S., Cechinel, C.: Exploring user-based recommender results in large learning object repositories: the case of MERLOT. In Manouselis, N, Drachsler, H, Verbert, K, Santos, OC (eds) Proceedings of the 1st Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2010), Procedia Computer Science, 1(2):2859-2864 (2010)

19. Verbert, K., Manouselis, N., Xavier, O., Wolpers, M., Drachsler, H., Bosnic, I., Duval, E., Context-aware recommender Systems for Learning: a Survey and Future Challenges, IEEE Transactions on Learning Technologies, https://lirias.kuleuven.be/handle/123456789/338644 (2012)

A Trust-based Social Recommender for Teachers

Soude Fazeli, Hendrik Drachsler, Francis Brouns, Peter Sloep

Open Universiteit in the Netherlands, Centre for Learning Sciences and Technologies,
Soude.fazeli@ou.nl, hendrik.drachsler@ou.nl, francis.brouns@ou.nl,
peter.sloep@ou.nl

Abstract. Online communities and networked learning provide teachers with social learning opportunities to interact and collaborate with others in order to develop their personal and professional skills. In this paper, Learning Networks are presented as an open infrastructure to provide teachers with such learning opportunities. However, with the large number of learning resources produced everyday, teachers need to find out what are the most suitable resources for them. In this paper, recommender systems are introduced as a potential solution to address this issue. Unfortunately, most of the educational recommender systems cannot make accurate recommendations due to the sparsity of the educational datasets. To overcome this problem, we propose a research approach that describes how one may take advantage of the social data which are obtained from monitoring the activities of teachers while they are using our social recommender.

Keywords. Learning Network, recommender system, teacher, social data, social networks, sparsity, trust

1   Introduction

The Internet provides teachers with a social space to interact and access resources in the form of either knowledge content or knowledgeable people outside their school [28], [13]. Online learning communities and networked learning are increasingly accepted by teachers as opportunities to continuously develop their personal and professional skills [11], [7]. Learning Networks (LN) are online social networks that follow the main goal of professional online communities for lifelong learners such as teachers, who need continuous support and guidance to develop themselves both personally and professionally [29]. Learning Networks can provide teachers with an open infrastructure not only to share, annotate, rate and tag content, but also to exchange knowledge and experience with the other members of the LN. Learning from others in a social context is a promising form of learning, which motivates learners to continuously learn and exchange knowledge. Research has shown the positive effects of social learning [31], [8], [4]. In this paper, we discuss how one may take advantage of LNs as an infrastructure to support teachers as lifelong learners.

With the increasing amount of user-generated content produced everyday in the form of learning resources, videos, discussion forums, blogs, etc., it becomes ever more difficult for teachers to find the most suitable content for their needs. Moreover,

to support social learning, teachers need to be supported to find the most suitable people who can help them to learn more effectively by sharing knowledge and experiences [31]. Generally speaking, recommender systems have emerged as a practical approach to provide a user with the most suitable objects based on their past behaviour. Recommender systems have become popular because of their successful applications in the e-commerce world such as in Amazon[1] and eBay[2]. Fortunately, they can be adjusted to be used also in the educational domain [10], [21].

In general, recommender systems suggest items to a target user. They do so based on the similarity between an item's content description and the user's preferences model (content-based); or they measure similarity between user profiles to predict an item's rating for a target user based on the rating history of the users who are similar to the target user (collaborative filtering). In this research, we take advantage of collaborative filtering methods as we mainly focus on the interactions and collaborations between teachers within a social environment. However, it is too difficult to compute similarity of user profiles when there is no common set of ratings between the users or when there are too little rating data available; this is known as the *sparsity problem*. Educational datasets suffer from this problem more often than commercial datasets [32]. Therefore, we need to find ways to overcome the sparsity problem in educational datasets if it is our aim to enhance the performance of recommender systems in learning. Social trust has been introduced to many recommender systems as a response to the sparsity problem [14], [36], [16], [19], [17]. Ziegler and Golbeck [36] show a strong connection between trust and similarity between users. In general, users prefer to receive recommendations from the people they trust. Golbeck [14] shows that trust captures not only simple overall similarity between users but also other features of the profile similarity

In teachers' communities, teachers can perhaps be supported to find trustworthy resources as proxies for reliable sources of information. Such trustworthy resources enable teachers to feel more comfortable to share and interact within a closed and trustful community. To achieve this, we follow a trust-based recommender system proposed by [12] to create trust relationships between users based on the rating information of user profile and item profile. Fazeli et al. [12] proposed a concept called T-index to measure trustworthiness of users in order to improve the process of finding the nearest neighbours. The T-index is inspired on the H-index, which is used to evaluate the publications of an author. The higher the T-index value of a user, the more trustworthy the user becomes. Fazeli et al. showed how the T-index improves structure of a generated trust network of users by creating connections to more trustworthy users [12]. They computed the trust values between users based on the ratings users gave to the items in their system. Although ratings' information is one of the examples of users' activities within a social environment, other social activities of users also can be worthwhile and should not be ignored up front. In general, the social activities of users describe each action of users within a social environment, for instance browsing a Web page, bookmarking, tagging, making a comment, giving rating, etc.

---

[1] http://www.amazon.com
[2] http://www.ebay.com

We refer to the data that comes from the social activities of users, as "social data". In this research, we aim to enhance the existing trust-based recommender of Fazeli et al. [12] by social data which are obtained from monitoring the activities of teachers while they are using our social recommender.

Therefore, the first research question is:

RQ1: How can the sparsity problem within educational datasets be solved by using inter-user trust relationships which originally come from the social data of users?

Moreover, we aim to investigate the evolution of LNs while we collect social data from users. Therefore, we need to study the structure of LNs for teachers to show how using social data can help to cluster teachers more precisely and as a result to find the most suitable content or people for their needs. So, the second research question is:

RQ2: How can teachers' networks be made to evolve by the use of social data?

In the following section, we present the research methodology used to address these two questions.

2 Proposed research

Our main objective is to support teachers to find the most suitable content or people and do so more effectively. The idea is that through finding suitable peers and content they will be better able to develop their personal and professional skills.

In order to achieve this goal, we follow the methodology described by [22] for recommender systems in TEL. We extend the methodology by first conducting an interview study with teachers. The research work, therefore, consists of four steps: 1. Requirement analysis (literature review and interview study), 2. Dataset-driven study , 3. User evaluation study, 4. Pilot study. We will describe each step in terms of its main goal, used methods, and the expected outcomes, in the following subsections.

2.1 Requirement analysis (literature review and interview study)

- **Goal.** Besides a literature study on the issues and challenges teacher often face, we organized interview group sessions with teachers and collected information from them in order to investigate their main needs and requirements.
- **Method.** The interview group session was conducted using the nominal group technique (NGT) [9]; the session took almost 2 hours and 45 minutes. The participants were 18 teachers (novices, experts, mentors and supervising teachers) from different schools in the Limburg area, the Netherlands, invited by Fontys Hogeschool.
- **Description.** During the session, the participants were asked to write down their ideas about the following question: "What kind of support do you need to provide innovative teaching at your school?" Then, we asked them to discuss the ideas gen-

erated and finally, to rank the ideas based on a five-point Likert scale (1 for the least interesting idea and 5 for the most interesting one). The teachers generated 121 ideas in total. The clustering was done during the session by the researchers (the alternative, to have the teachers do it, was rejected because of time limitations). After the session, we invited the teachers to cluster the ideas in a Web-based application called Websort[3]. The data are still being analysed.

- **Expected outcomes.** An inventory of teachers' needs and requirements will be the outcome of this step. This inventory list will be used to as an input to design a recommender system which suits teachers' needs the best.

2.2 Dataset-driven study

- **Goal.** The main goal is to validate the framework we propose which presents the important characteristics of a recommender system to be designed for teachers. We will elaborate the framework in details in Section 3.
- **Method.** An offline empirical study of different algorithms on a selected set of representative datasets is to be conducted. The offline experiments (data study) on educational datasets will be in terms of the popular metrics often used to evaluate the performance of recommender systems.
- **Variables to be measured.** Prediction accuracy and coverage of the generated recommendations are the variables to be measured in this step.
- **Description.** Based on the literature review and the interview study, we present a framework to identify the suitable recommender systems' strategies to be applied for our target users which helped us to make an effective selection of the available educational datasets. The selected educational datasets for teachers to be studied are TravelWell [33], MACE[4], Organic.Edunet[5], TELeurope[6], OpenScout[7], digischool[8] and eTwinning[9].
- **Expected outcomes.** Initial results will indicate which of the recommender system algorithms suits teachers best and if the trust-based recommender system can help to deal with the sparse data in the used datasets.

2.3 User evaluation study

- **Goal.** The goal is to study usability of the prototype by evaluating users' satisfaction.
- **Method.** The experiment will be done by a questionnaire through which end-users will be asked to provide feedback on the prototype.

---

[3] http://uxpunk.com/websort/
[4] http://portal.mace-project.eu
[5] http://portal.organic-edunet.eu
[6] http://www.teleurope.eu/
[7] http://www.openscout.net
[8] http://www2.digischool.nl/leerling/vo
[9] http://www.etwinning.net/en/pub/index.htm

- **Variables to be measured.** User evaluation will be in terms of interestingness (how much the end-users find the recommended content or people interesting) and value-addedness (how recommended content or people can help users to gain new knowledge or improve their current knowledge) [32].
- **Description.** Based on the outcomes, the prototype will be customized and improved so as to be able to deploy an improved release in a pilot study.
- **Expected outcomes.** Initial feedback by end-users on usability of the prototype is the outcome we expect.

2.4 Pilot study

- **Goal**. We aim to deploy the final release to test it under realistic and normal operational conditions with the end-users.
- **Method.** We compare the performance of a proposed recommender system based on our presented framework with classical collaborative filtering algorithms. Furthermore, we aim to study the structure of the teachers' networks to investigate how networks of teachers will evolve by use of social data. To evaluate the effectiveness of the proposed recommender system, we will compare the results in terms of total number of learning objects which have been visited, bookmarked, rated, etc. for two groups of users:
  - Those who are aided by recommender systems to access learning objects
  - Those who access learning objects directly from the repository, without the help of a recommender system.
- **Variables to be measured.** We will measure prediction accuracy and coverage of the generated recommendations, effectiveness in terms of total number of visited, bookmarked, or rated learning objects, as well as Indegree distribution used to study how the structure of the networks changes. For a node on a network, Indegree describes the number of coming edges (or relationships) to the node.
- **Expected outcomes.** We expect to obtain empirical data on prediction accuracy and coverage, to validate whether our proposed recommender system outperforms the classical CF algorithms. Another outcome will be the visualization of teachers' networks, to show how the network's structure evolves when relying on social data.

Having given an overview of the research method, we will now present the state-of-the-art in recommender systems to allow us to explore the characteristics which should be taken into account to design a suitable recommender system for teachers.

3 State-of-the-art

Several reviews exist which detail how to study and classify recommender systems in terms of recommendation techniques, tasks, delivery mode, etc [22]. However, each of these reviews focuses only on some of the dimensions to classify recommender systems and none of them present an integrated framework for the classification of

recommender systems. Manouselis and Costopoulou [22] propose a framework to categorize the dimensions of recommender systems, which were identified in the previous studies. We will use this framework to investigate the characteristics that should be considered to design a recommender system for teachers. As shown in Figure 1, the proposed framework consists of five main categories of characteristics: Supported tasks, User model, Domain model, Personalization, and Operation. In the rest of this section, we will introduce each of the characteristics briefly and we will conclude how the resulting framework could be applied to a recommender system for teachers.
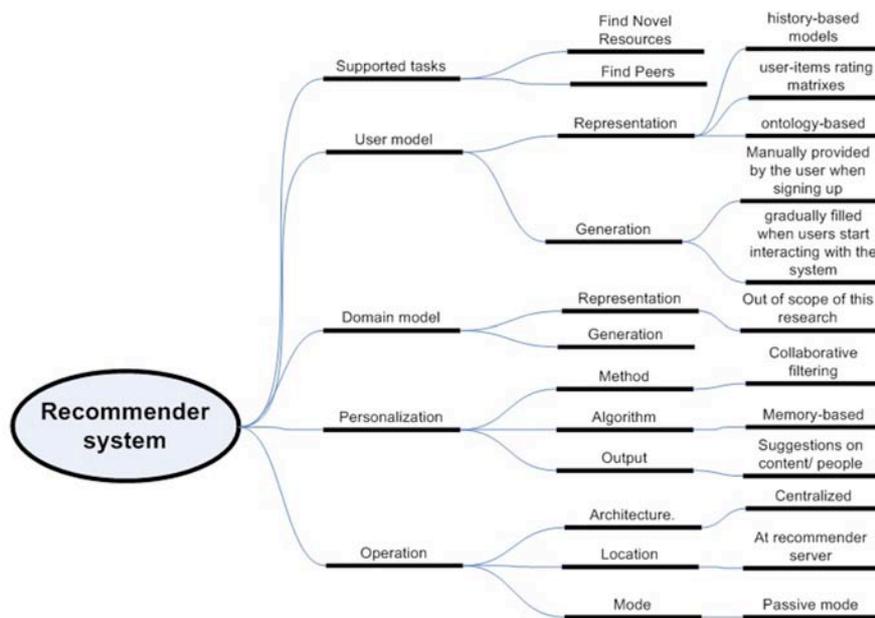
Fig. 1. A proposed social recommender system for teachers

3.1 Supported Tasks

As mentioned before, teachers need to keep informed about the resources which can inspire them to deal with the issues they face in their job. So, we aim to support teachers to *Find Novel Resources* which are suitable for them based on their profile history. Most of the recommender systems in the educational domain have been designed to support this task [25], [18], [24], [31], [10]. For more examples, refer to the book by Manouselis et al. [22]. Moreover, teachers need to be supported to *Find Peers* who can be trusted to share their concerns with them and to receive help from them, so-called trustworthy peers. According to an extensive overview of the recommender systems in the Technology Enhanced Learning (TEL) field provided by Manouselis et al. [22], only few of the recommender systems aim to support this task [25], [1].

3.2 User model

We represent user profiles for teachers by history-based models and user-item matrix which mainly focus on the past activities of the users such as ratings information [25], [18], [10], [22]. Furthermore, we aim to create user profiles based on ontology to provide more interoperability and openness among different platforms. Therefore, we are going to use ontology to model the relationships between teachers in social networks [12], [14]. The user profiles for teachers are generated based on the information provided by the users when they themselves fill in a registration form with their personal information (name, surname, email, etc.) and professional information (teaching subject, interests, background knowledge, etc.). We refer to this part of the user profile as *static data* as it can be edited manually by the users anytime they want to. The other part of the user profiles contains *recommendation data.* It will be updated by our system as soon as teachers start interacting with the system. Since our main objective is to support teachers with a recommender system in the educational domain, we have to take into account the teachers' characteristics. So, to create a user model for teachers, we need to consider both actions of teachers and context variables in the TEL field [34]. Verbert et al. [34] describe the main characteristics that are to be considered for users in an educational context, such as knowledge level, interests, goals and tasks, and background knowledge, in addition to the data regarding users' actions in terms of type and result of actions and the context in which an action has been taken.

As indicated, we intend to take advantage of social data of users to deal with the sparsity problem. To do so, we keep track of users' actions within our system, so-called social activities, when they rate, tag, and bookmark content. In this way, the recommendations will be generated and improved based on the recorded actions of teachers while they interact with our system. As mentioned before, social data originally comes from these recorded actions of users (teachers). To capture their social data, we need to follow a standard specification to store and maintain their actions. Several standard specifications to describe social data of users and guarantee their interoperability exist. They are:

- **FOAF.** The FOAF (Friend-of-a-Friend) vocabulary [3] describes user's information and their social connections through concepts and properties in form of an ontology using Semantic Web technologies [14]. The FOAF Vocabulary describes personal information and social relationships. The FOAF Vocabulary shows basic information of users (FOAF Basics) such as name, surname and also personal information about the people that a user "knows" and its interest area (Personal Info). In this research, we could extend the FOAF ontology to describe user profiles and to model the social relationships between teachers by the concept of *FOAF:agent.*
- **CAM.** Contextualized Attention Metadata (CAM) is a format to capture observations about users' activities with any kind of tool [35]. A CAM schema aims to store whatever has attracted users' attention while the users are working with the tool. It also stores users' interaction with the tool such as rating, tagging, etc. A CAM schema records an event and its details when a user performs an action within a tool. The metadata stored in the CAM format describe all type of users' feedback and, therefore, can be used to make recommendations for the users. We could

make use of the CAM schema to capture the users' activities within our system and as a result, to extract the social data of users in order to create user profiles.

Annotation scheme. In the context of Organic.Edunet[10], Manouselis and Vuorikari [20] developed a model to represent and store users' feedback including rating, tagging, reviewing, etc. in a structured, interoperable and reusable format. This model is based on the CAM format. Manouselis and Vuorikari called it an annotation scheme and proposed it as a structured and interoperable format to be used to transfer the social data of users between heterogeneous systems. We could take advantage of the annotation scheme to describe social data of users within our system.

3.3   Domain model

Items to be presented to teachers need to be represented somehow and need to be generated before they can be presented. This task is out of scope for the present research project. It will, parenthetically, be taken up by the Open Discovery Space project[11] which aims to represent learning objects in the form of an integrated object repository containing several collections of learning objects which are hosted by the ARIADNE[12] infrastructure.

3.4   Personalization

Method. In general, and as we already pointed out in Section 1, there are two main types of algorithms used in recommender systems: content-based and collaborative filtering. Content-based algorithms compare the description of an item with representations of users' interests. Amazon is a good example of such a recommender system, which provides a so-called 'favorites' feature to represent the preferred items by users. The 'favorites' are introduced as the content-based part of user profiles, which are either manually provided by a user or are calculated based on purchase history of the user [23]. As content-based algorithms make recommendations for a user only based on the user's interests individually, the user is less likely to find novel items which might be interesting to the user [6]. Collaborative Filtering (CF) is another type of recommender systems which purely depends on opinions and ratings of users instead of actual content descriptions. CF algorithms search for like-minded users that are introduced as neighbourhoods and they predict an item's rating for a target user based on collected ratings of the user's neighbours [27]. They recommend a target user top-N recommended neighbours and/or items. Traditional CF algorithms form the neighbourhoods based on similarity between user profiles [25], [24], [10], [21], [33].

As mentioned before, traditional CF algorithms suffer from the sparsity problem if too little rating information is available to compute similarity between users. Social

---

[10] http://portal.organic-edunet.eu/
[11] Open Discovery Space is a 7th framework European project partly sponsored the presented research work in this document.
[12] http://www.ariadne-eu.org/repositories

trust has emerged as a solution to the sparsity problem in many recommender systems [14], [36], [16], [19], [17]. In the research area of recommender systems, trustworthy users have been introduced as like-minded users and thus, trust originates from similarity between users. However, assuming that trust is transitive (if A trusts B and B trusts C, then A trusts C), we may find a relationship between two users who have no common set of items but do have friends in common. Suppose we have two users: Alice and Carol who have no rated set of items in common. Therefore, it is not possible to compute similarity between them. As a result, there will no direct relationship between Alice and Carol even though they are already indirectly connected through another user whose name is Bob. In this case, the inter-user trust phenomena helps us to infer a relationship between Alice and Carol through their common friend Bob because if Alice trusts Bob in his recommendations on papers and Bob also trusts Carol in the same way then, Alice can trust Carol in her recommendations on papers. This is how we define "trust" in this research work. Therefore, we form neighbourhoods based on the trust relationships between users and we introduce the top-N neighbours, commonly used in CF, as the most trustworthy users for a target user. To do so, we are going to adjust an existing trust-based recommender system proposed by Fazeli et al. [12] to make it suitable for an educational setting, particularly for teachers. Furthermore, we aim to take advantage of social data of users described in Section 3.2, to boost the performance of our proposed recommender system for teachers.

Type and Technique. CF methods are often categorized according to type or technique. Type refers to memory-based and model-based algorithms; and technique refers to user-based, item-based, and attribute-based approaches [26], [28]. Model-based algorithms use probabilistic approaches to develop model of a user based on the user's history and profile. Examples of model-based algorithms are Bayesian networks, neural networks, and algebraic approaches such as eigenvectors [16]. Although these algorithms are faster than memory-based algorithms, they require a full set of users' preferences to develop user models; such a set is often not available. Moreover, model-based algorithms are often very costly for learning and updating phases. Instead, memory-based algorithms are quite straightforward to use. They find correlations between users based on statistical techniques for measuring similarity such as Pearson correlations or Cosine similarities [2]. In this research, we aim to use memory-based CF algorithms to recommend teachers the most suitable content or people, based on the user-based techniques which focus on the similarity between users in order to make recommendations [28].

Output. Most of the recommender systems generate recommendations in the form of suggestions on content or people, or sometimes ratings [25], [10], [1]. Another common output of recommender systems is predictions of a rating value that a user would give to an item [28], [33].

3.5 Operation

In this research, we intend to follow a centralized architecture, in which a central recommender server provides access to a single learning object repository. The recommendations are to be made at the recommender server (location) and are to be sent to the teachers as part of natural interactions of the users within our system, for example when the user browses a page or rates an item. In this way, teachers do not need to ask to receive recommendations explicitly (passive mode) [28].

4 Conclusion and further work

In this paper, we described why teachers need to be supported to find the most suitable content or people for their needs and we introduced recommender systems as a potential solution to address this issue. We also argued that we need to overcome the sparsity problem when we aim to enhance the performance of recommender systems in the educational domain and particularly for teachers. Therefore, we presented our research questions and research method that mainly focus on a solution to tackle the sparsity problem. As part of our proposed research based on the literature study, we proposed a framework that explores the main characteristics required to design a recommender system approach that suits teachers' needs the best. To validate this framework, we already started to set up an offline empirical study to test different algorithms of recommender systems on the selected datasets. As for the requirement analysis, an interview study has been conducted for 18 teachers from the Netherlands who already have been invited to cluster their ideas by Websort, following up the group session we had with them (described in Section 2.1). Furthermore, we took advantage of the Open Discovery Space Summer School in Greece, in July 2012 to involve more teachers in the Websort study. As a result, we now have an extensive analysis of the requirements for teachers all over the Europe. We are currently investigating the data and will present outcomes of the study in a special issue of the RecSysTEL workshop that will be published by Springer.

References.
1. Beham G, Kump B, Ley T, Lindstaedt S. (2010) Recommending knowledgeable people in a work-integrated learning system. Proceedings of the 1st Workshop on Recommender

Systems for Technology Enhanced Learning (RecSysTEL 2010), Volume 1, Issue 2, 2010, Pages 2783-2792

2. Breese J. S., Heckerman D., and Kadie, C. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI 1998), pages 43–52, 1998.

3. Brickley D., Miller, L. (2004) FOAF vocabulary specification. Namespace Document, pages 331–370

4. Brown J. S., Adler R. P. (2008). Minds on Fire: Open Education, the Long Tail, and Learning 2.0. EDUCAUSE Review, 43(1), 16-32.

5. Burke R. Hybrid Web Recommender Systems. In Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.). The Adaptive Web: Methods and Strategies of Web Personalization., Lecture Notes in Computer Science, Vol. 4321, pp. 377-408, Springer, Berlin-Heidelberg, 2007.

6. Cantador I., Bellogín A., Vallet D.: Content-based recommendation in social tagging systems. RecSys 2010: 237-240

7. Daly A. J., Moolenaar N. M., Bolivar J. M., Burke P. (2010) Relationships in reform: the role of teachers' social networks. Journal of Educational Administration, 48(3), 359-391. doi: 10.1108/09578231011041062

8. Dawson S. (2008). A study of the relationship between student social networks and sense of community. Educational Technology & Society, 11(3), 224-238.

9. De Vries F., Kester L., Sloep P., Van Rosmalen P., Pannekeet K., Koper R. (2005). Identification of critical time-consuming student support activities in e-learning. Research in Learning Technology (ALT-J), 13(3), 219-229.

10. Drachsler H., Pecceu D., Arts T., Hutten E., Rutledge L., Van Rosmalen P., Hummel H., Koper R.: ReMashed - Recommendations for Mash-Up Personal Learning Environments. In: Cress, U., Dimitrova, V., Specht, M. (eds.): Learning in the Synergy of Multiple Disciplines, EC-TEL 2009, LNCS 5794, Berlin; Heidelberg; New York: Springer, pp788-793, 2009

11. Duncan-Howell J.: Teachers making connections: Online communities as a source of professional learning, British Journal of Educational Technology, Volume 41, Issue 2, pages 324–340, 2010

12. Fazeli S., Zarghami A., Dokoohaki N., Matskin M. (2010) Elevating Prediction Accuracy in Trust-aware Collaborative Filtering Recommenders through T-index Metric and TopTrustee lists, 300-309. In JOURNAL OF EMERGING TECHNOLOGIES IN WEB INTELLIGENCE 2 (4).

13. Fox A., Wilson E., & Deaney R. (2011). Beginning Teachers' Workplace Experiences: Perceptions of and Use of Support. Vocations and Learning, 1-24. doi: 10.1007/s12186-010-9046-1

14. Golbeck J. Computing and applying trust in web-based social networks. PhD thesis, University of Maryland at College Park, College Park, MD, USA, 2005.

15. Hirsch J. E. An index to quantify an individual's scientific research output. PNAS, 102(46):16569–16572, November 2005.

16. Kamvar S., Schlosser M., Garcia-Molina H. (2003) The eigentrust algorithm for reputation management in p2p networks. In Proceedings of WWW2003. ACM, 2003.

17. Lathia N., Hailes S., Capra L. Trust-based collaborative filtering. In IFIPTM 2008: Joint iTrust and PST Conferences on Privacy, Trust management and Security, page 14, Department of Computer Science, University College London, London, UK, 2008.

18. Lemire D., Boley H., McGrath S., Ball M. (2005). Collaborative Filtering and Inference Rules for Context-Aware Learning Object Recommendation. International Journal of Interactive Technology and Smart Education, 2(3).

19. Massa P., Avesani P., (2007) Trust-aware recommender systems. RecSys '07 Proceedings of the ACM conference on Recommender systems, Pages 17-24, ACM New York, USA

20. Manouselis N., Vuorikari R. (2009) What If Annotations Were Reusable: A Preliminary Discussion. ICWL 2009: 255-264

21. Manouselis N., Vuorikari R., Van Assche F. (2010) Collaborative recommendation of e-learning resources: an experimental investigation. JCAL, 26(4):227242.

22. Manouselis N., Drachsler H., Verbert K., Duval E. (2012). Recommender Systems for Learning book, Springer, to appear.

23. Pazzani M. J., Billsus D. (2007). Content-based recommendation systems. In P. Brusilovsky, A. Kobsa & W. Nejdl (Eds.), The adaptive web: methods and strategies of web personalization (Vol. 4321, pp. 325-341): Springer-Verlag.

24. Rafaeli S., Dan-Gur Y., Barak M. (2005) Social Recommender Systems: Recommendations in Support of E-Learning. International Journal of Distance Education Technologies, 3(2), 29-45.

25. Recker MM, Walker A, Lawless K (2003) What do you recommend? Implementation and analyses of collaborative information filtering of web resources for education. Instructional Science, 31(4/5):299–316.

26. Sarwar B. M., Karypis G., Konstan J. A., and Riedl J. (1998) Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International World Wide Web Conference (WWW 2001), pages 285–295.

27. Schafer JB, Frankowski D, Herlocker J, Sen S (2007) Collaborative Filtering Systems. In Brusilovsky P, Kobsa A, Neidl W. (eds) The Adaptive Web: Methods and Strategies of Web Personalization, Lecture Notes in Computer Science, Vol. 4321, pp. 291-324, Springer-Verlag, Berlin Heidelberg New York.

28. Schafer JB, Konstan JA, Riedl J (2001) E-commerce Recommendation ApplicationsData Mining and Knowledge Discovery, 5:115-153.

29. Schuck S., Getting help from the outside: developing a support network for beginning teachers, The Journal of Educational Enquiry, Vol 4, No 1 (2003)

30. Sloep P. (2009). Social Interaction in Learning Networks. In R. Koper (Ed.), Learning Network Services for Professional Development (pp. 13-15). Berlin, Germany: Springer Verlag.

31. Tang T.Y., McCalla G.I., "Smart Recommendation for an Evolving E-Learning System: Architecture and Experiment", International Journal on E-Learning, 4 (1), 105-129, 2005.

32. Tiffany, Y. Tang, & Gordon, McCalla. The Pedagogical Value of Papers: a Collaborative-Filtering based Paper Recommender. Journal of Digital Information; Vol 10, No 2 (2009)

33. Vassileva J. (2008). Toward Social Learning Environments. *IEEE Transactions on Learning Technologies, 1*(4), 199-214.

34. Verbert K., Drachsler H., Manouselis N., Wolpers M., Vuorikari R., Duval E., Dataset-driven research for improving recommender systems for learning, Proceeding LAK '11, Proceedings of the 1st International Conference on Learning Analytics and Knowledge, Pages 44-53, ACM New York, NY, USA, 2011

35. Verbert K, Manouselis N, Drachsler H, Duval E. (to appear) Dataset-driven Research to Support Learning and Knowledge Analytics. Educational Technology & Society, Special Issue on Learning and Knowledge Analytics, to appear.

36. Wolpers M., Najjar J., Verbert K., & Duval E. (2007). Tracking Actual Usage: the Attention Metadata Approach. Educational Technology & Society, 10 (3), 106-121.

37. Ziegler C.-N., Golbeck J. (2006) Investigating Correlations of Trust and Interest Similarity. Decision Support Services.

Sentiment Analysis: A tool for Rating Attribution to Content in Recommender Systems

Antonis Koukourikos[1], Giannis Stoitsis[2,3], Pythagoras Karampiperis[1]

[1]Software and Knowledge Engineering Laboratory, Institute of Informatics and Telecommunications, National Center for Scientific Research "Demokritos"Agia Paraskevi Attikis, P.O.Box 60228, 15310 Athens, Greece
[2]Agro-Know Technologies, Grammou 17 Str., Vrilissia Attikis, 15235 Athens, Greece
[3]Universidad de Alcalá, Pza. San Diego, s/n - 28801 Alcalá de Henares, Madrid, Spain

Abstract. Collaborative filtering techniques are commonly used in social networking environments for proposing user connections or interesting shared resources. While metrics based on access patterns and user behavior produce interesting results, they do not take into account qualitative information, i.e. the actual opinion of a user that used the resource and whether or not he would propose it for use to other users. This is of particular importance on educational repositories, where the users present significant deviations in goals, needs, interests and expertise level. In this paper, we propose the introduction of sentiment analysis techniques on user comments regarding an educational resource in order to extract the opinion of a user for the quality of the latter and take into account its quality as perceived by the community before proposing the resource to another user.

Keywords: Recommender Systems, Educational Repositories, Sentiments Analysis, Qualitative Analysis

1   Introduction

Recommender Systems are of particular importance within social environments, where users share access to a common set of resources. The variability of crucial user characteristics, like their background, their special interests, their degree of expertise, pose interesting issues in terms of proposing a resource that is interesting, useful and comprehensible to a particular user.

Collaborative filtering approaches based on explicitly given user ratings do not always reflect the differentiation between the various criteria that apply to a resource and the weight that the users give to each criterion. On the other hand, techniques that examine access patterns may suffer from the appearance of stigmergy phenomena. The visibility of a resource, or even more elaborate features like the time spent in a resource, the amount of downloads etc. are not directly connected to its quality or suitability. Hence, the examination of access and use patterns can lead to poor recommendation that will be further propagated due to the users continuing to follow previously defined paths within the repository of available content.

In this context, we propose the exploitation of user generated comments on the resources of a repository of educational content in order to deal with the lack of explicit ratings and discover qualitative information related to a specific resource and the impressions it left to the users that accessed it. To this end, we applied sentiment analysis to comments on educational content and examined the accuracy of the results and the degree to which they reflect user satisfaction.

The rest of the paper is structured as follows: we provide a brief review of the sentiment analysis in Section 2. We present the four algorithms that we aim to implement and examine for the Organic.Edunet recommendation system in Section 3. Section 4 describes the experimental setup and the results for the first of the proposed approaches. We conclude with our conclusions so far and report on the intended next steps.

2   Related Work

Recommender systems, particularly using collaborative filtering techniques, aim to predict the preferences of an individual (user/ customer) and provide suggestions of further resources or entities (other users of the same system, resources, products) that are likely to be of interest. The usage of recommender systems is widely spread in e-commerce environments [1] but the general principle is applica-

ble to multiple and diverse environments. In the case of TEL, multiple solutions have been proposed and examined [2, 3]. Due to the particularities of the domain, some of the most common algorithms for collaborative filtering have been shown to struggle in the setting of a learning object repository [4, 5]. As mentioned, the presented techniques are examined in order to be incorporated in a recommender system over a social platform that provides access to educational content. Linguistic techniques, such as sentiment analysis, can be of use for alleviating some of the drawbacks of traditional algorithms in terms of differentiating users belonging in different audiences (e.g teachers from students) and bypassing the need for explicit ratings (via a star system).

Sentiment analysis regards extracting opinion from texts and classifying it into positive, negative or neutral valence [6]. Work on the field focuses on two general directions; lexical approaches and solutions using supervised machine learning techniques.

Lexical approaches rely on the creation of appropriate dictionaries. The terms present in the dictionary are tagged with respect to their polarity. Given an input text, the presence of dictionary terms is examined and the overall sentiment of the text is computed based on the existence of "positive" and "negative" terms within it. Despite its simplicity, the lexical approach has produced results significant better than "coin-toss" [7, 8, 9]. The way of constructing the lexica that are used for sentiment analysis is the subject of several works. In [10] and [11] the lexicons comprised solely adjective terms.

The usage of pivot words (like "good" and "bad") and their association with the target words is also a frequently met approach. In [9] and [12], the minimum path between each target word and the pivot terms in the WordNet hierarchy was calculated in order to determine the polarity of the term and its inclusion in the dictionary. In [8], the authors executed search queries with the conjunction of the pivot words and the target word given as input. The query that returned the most hits determined the polarity of the given word.

Machine learning techniques focus on the selection of feature vectors and the provision of tagged corpora to a classifier, which will be used for analysing untagged corpora. The most frequent routes for choosing the feature vectors are the inclusion of unigrams or n-grams, counting the number of positive/negative words, the length of the document etc. The classifiers are usually implemented as a Naive Bayes classifiers or as Support Vector Machines [9, 13]. Their accuracy is dependent on the selection of the aforementioned feature vectors, ranging in the same space as the lexical approaches (63%-82%).

3 Algorithms under Analysis

For our experiments, we aim to examine the following sentiment analysis algorithms and evaluate their performance in order to deploy the most suitable for a repository of educational content.

The fact that we are dealing with user generated content drives us to take into account its unstructured nature and the potential unbalanced distribution it may present. This gives rise to the fact that our training set may be unbalanced and therefore learning may not be able to cope with such diversity in the number of instances per class. Hence, these properties require simulating sentiment representations onto which the input text will be mapped, since sentiment prediction calls for predefined knowledge. Therefore, we focus on lexical approaches for capturing the polarity expressed in a comment. Specifically, we produced implementations of the following algorithms.

3.1 Affective Terms Frequency

Rather small documents or text chunks that carry a certain kind of sentiment polarity have been found to present that valence throughout the text, or in most parts of it. For example, in tweets we observe cases like: "Don't you just love this camera? It's great!" In such a piece of discourse, the probability of tracing negative terms is rather low.

This observation gives rise to determining the affective term frequency that appears in cases as the above mentioned. In order to capture the overall sentiment expressed in such inputs, we proceed as follows:

The algorithm receives as input the text to be processed and two lists of affective terms, one of positive and one of negative valence.

For every word of the text to be processed, we examine whether it is mapped on either of the two lists [14]. If it matches an entry of in either of them, a corresponding value gets incremented by 1. After having traversed the whole text, we compare the two sums and the one with the highest value is con-

sidered to be the dominant one and the respective valence is attributed to the input text [15]. If they are equal or no sentiment is detected, the text is considered to carry neutral sentiment.

In specific, we calculate:

$$\Delta = \sum_{i=0}^{n-1} x_i - \sum_{j=0}^{n-1} y_j \ (1)$$

where, Δ is the absolute difference between the positive and negative sums, x_i is each instance in the vector representing the positive valence, y_j is each instance in the vector representing the negative valence and n is the number of words the input text may contain.

If $\Delta > 0$, the sentiment is positive, if $\Delta < 0$, the sentiment is negative, else the sentiment is neutral.

3.2 Weighted Affective and Domain Term Frequency

Sentiment attribution varies according to the domain(s) the text to be examined belongs to. For example, the term "small" is considered to be negative, due to its connotation with insufficiency. However, in the domain of computer hardware and mobile devices, its valence is shifted and it's mostly attached with positive opinions. For instance, in a product review or in a forum, we mostly come across statements like: "The iPad 2 is amazing! It's so small and light I can take it with me everywhere". In order to be able to identify such assertions, we may need to add to every affective term a frequency value that will determine how positive, negative or neutral it is [16].

As a deduction, the same principle applies for all the text terms (excluding stop words). The reason for such a precaution is that specific terms that don't appear to influence text valence, in general, change their sentiment in regard to a specific context. For instance, the verb "watch" doesn't seem to present any specific sentiment. However, in movie reviews, we come across comments like the following: "You should definitely watch this movie". In this example, a positive opinion can be detected neither in a word, nor in an idiom or irony or any other discourse schema. Yet, the sentence bears it.

One may argue that this doesn't guarantee any credibility, since another user comment of e.g. IMDB could be: "You should definitely not watch this movie". Nevertheless, the reason the sentiment expressed in the second comment is opposite to the first one is the presence of the valence shifter: "not" and not of an affective term or discourse schema denoting negative valence. This would be the case if the two comments would be the following respectively: "I enjoyed this movie!" versus "I did not enjoy this movie". As we can tell, in either case, the sentiment of the affective term is reversed, irrespectively of the term belonging to a list of affective terms, i.e. the term "enjoy", or not belonging to a list of affective terms, i.e. the term "watch".

The algorithm receives as input the text to be processed and three hash tables of affective terms and their frequency, one of positive, one of negative and one of neutral valence.

The hash-table lists of affective terms are constructed as follows: A set of domain specific terms is built [17, 18]. Depending on the domain, the corpus may consist of product reviews, critiques on results of intellectual effort (music, movies) or more formal documents like questionnaires, review forms etc. Human annotators examine the polarity of this content and the corpus is partitioned in three subcorpora, one consisting of positive, one of negative and one of neutral terms. For every word in the sub-corpus of positive annotated texts, we attribute its frequency in this specific corpus. The same procedure takes place for the other two corpora. Sub-sampling may have taken place where necessary.

As a result, if a term appears in all three sub-corpora, it receives three values, one attributed to each of them respectively. If it appears in two or in one it receives the score it has been assigned within this corpus. The reason why the sum of positive, negative and neutral doesn't sum up to 1 is that, if, for example, word "nice" appears only in positive terms, it would be assigned weight value 1. At the same time, if, for example, word "excellent" again appears only in positive terms, it would be assigned weight value 1. Subsequently, such an attribution would oppose the current approach, aiming at measuring sentiment as per weighted frequencies. Such weighted frequencies get enhanced by predefined lists.

For positive, negative and neutral valence, three hash tables are created, containing the positive, the negative and the neutral lists respectively. For every word of the text to be processed, we examine whether it is mapped on either of the three hash tables. If a key in the hash table comprising the positive terms maps on the word in question, its value is added in a vector keeping the positive term frequencies. If the hash table comprising the negative terms also contains this word, its value is added in a

vector keeping the negative term frequencies and so on. After the whole document has been traversed, the values of each of the three vectors are summed up, the three sums are compared and the one with the highest value designates the sentiment of the respective post.

In specific, we calculate:

$$\Delta = \sum_{i=0}^{n-1} x_i \cdot w_i - \sum_{j=0}^{n-1} y_j \cdot w_j \ (2)$$

where, Δ is the difference between the positive and negative sums, x_i is each instance in the vector representing the positive valence of weight w_i, w_i is the value of the positive weight of instance x, y is each instance in the vector representing the negative valence of weight w_j and w_j is the value of the negative weight of instance y_i.

If $\Delta = 0$, the sentiment is neutral.

Else, if $\Delta > 0$, we calculate:

$$\Delta_1 = \sum_{i=0}^{n-1} x_i \cdot w_i - \sum_{k=0}^{n-1} z_k \cdot w_k (3)$$

where Δ_1 is the absolute difference between the positive and neutral sums, z_k is each instance in the vector representing the neutral valence of weight w_k and w_k is the value of the neutral weight of instance z_k. If $\Delta_1 = 0$, the sentiment is neutral, else it is positive.

Else, we calculate:

$$\Delta_2 = \sum_{j=0}^{n-1} y_j \cdot w_j - \sum_{k=0}^{n-1} z_k \cdot w_k (4)$$

where Δ_2 is the absolute difference between the negative and neutral sums. If $\Delta_2 = 0$, the sentiment is neutral, else it is negative.

3.3   Distance between Affective and Sentiment Targeted Terms

When we are called to tackle a more specific problem, e.g. a more targeted question in a more concrete domain, it is required that our processing is also more focused on the object, i.e. the entity or entities that represent it, and/or the domain towards which opinion is expressed. This way, we aim at capturing the entity characteristics that affect sentiment rendering. For example, we may come across a comment like: "I am discontent by the book I had read and I've found it rather useless, but this video is really great." The previous two approaches would classify this comment as being negative as regards the video, while the writer expresses positive opinion about it.

Moreover, comments, in specific, comprise in a lot of cases unstructured chunks and other abnormalities, such as emoticons, artificial words, e.g. "yeeeeees", punctuation mistakes, e.g. "What is that?????" and many more syntactic and grammatical deviations from standard language.

In consideration of such differentiations, we count the distance between the affective terms and the terms that are involved in the representation of the entity towards which sentiment is expressed. If sentiment is expressed towards more than one entity or entity representations, then all distances are counted recursively.

The algorithm receives as input the text to be processed, two lists of affective terms, one of positive and one of negative valence and the entity/entities towards which sentiment is expressed. In the rest of the paper we will also refer to these latter terms as "keyword(s)" for simplicity reasons.

The position of the entity towards which sentiment is expressed is tracked in the text to be processed. The words of the document to be examined are mapped onto the affective terms of the input lists. If the document contains an affective term, then its position in the text is tracked as well and we calculate the distance that separates them.

To be more specific, we detect whether the affective term precedes or follows the entity in question. After having located these two points in the text, we calculate the distance between them, in the substring that separates them [19]. This is based on word count versus character count, because in this

approach words are considered to be autonomous semantic meaningful units, unlike alphanumeric strings, regarded as self-contained units in graph-based approaches.

The above mentioned procedure takes place for all affective terms. In particular, for every positive term that appears in the input text, its distance from the entity in question is counted. After all positive terms have been checked, the smallest distance is kept to be compared with the respective smallest distance between the negative terms and the entity in question. If the two values equal to zero, or are equal, neutral sentiment is attributed. Otherwise, the post receives the sentiment represented by the smaller of the two values. If we have more than one entity representation, the same procedure is applied and again the shortest distance is taken as representing the sentiment of the writer.

In specific, we calculate:

$$\Delta = x - y \quad (5)$$

where Δ is the difference of the equation, denoting which of the two scores is higher, x is the minimum distance between the positive term and the key word and y is the minimum distance between the negative term and the keyword. If $\Delta > 0$, the sentiment is positive, else if $\Delta < 0$, the sentiment is negative, else, the sentiment is neutral.

3.4 Dependencies between Affective and Sentiment Targeted Terms

More formal documents tend to present a more consistent and accurate syntactic and grammatical structure, hence more concrete and concise textual forms. This characteristic restricts the number of alternatives we may have in expressing a certain meaning and therefore facilitates us to capture it. As a result, the better we are able to represent this structure, the closer we get in capturing the semantics it pertains [20].

In our approach, we are interested in detecting the syntactic dependencies between the keywords and the affective targeted term(s). In particular, when the sentiment analysis algorithm accepts a text as input, it accepts it attached to certain categories and/or the description of the educational material in question. As a consequence, our goal is to track the sentiment of the writer in relation to the material we are examining. For this reason, we process every sentence of the particular comment so as to identify whether a reference of the material is attached to an affective term. Syntactic Parsers provide the necessary tools to analyze the input text. An example is illustrated in Figure 1, where in the second line we can tell that, when referring to the relationship of two words we refer to words: "trustworthy, China" and the kind of relation that binds them is: "nsubj", namely: "China" is the subject of a verb and "trustworthy" is the attribute of the same verb.


```
currentSentence: China is trustworthy.
nsubj(trustworthy-3,china-1)
cop(trustworthy-3,is-2)
```

Fig. 1. Exemplary output of the Stanford Parser

The algorithm, thus, accepts as input the text to be processed, two list of affective terms (positive and negative), a list containing the keywords that define the text's context, a list of reporting verbs defining assertions, a list of verbs differentiating indirect speech to counterfactuals and a list of stop words.
The basic linguistic processing steps of sentence splitting and tokenization are performed, before obtaining the parse tree for each of the resulting sentences. The tokens are lemmatized in order to be mapped to the words included in the aforementioned lists e.g. the term "loved" should be lemmatized to "love", so as to be mapped on the corresponding entry of the list containing the positive terms. Next, we extract the dependencies between the input text lemmas and the words that are contained in our list. This procedure takes place in order to acquire those types of dependencies that will infer the sentiment polarity we will attribute, as we've shown in Figure 1.

After having tracked a keyword or keywords, we try to detect affective terms. If no such terms are tracked, then the sentence is considered as carrying neutral sentiment. This value is kept in a counter whose value is increased by 1 every time a sentence of neutral polarity is met. At the end of each post processing, all neutral values are being accumulated and point out a neutral sum, to be compared with the positive and negative ones.

On the other hand, if a sentiment-bearing word is tracked, we try to identify whether this sentiment word renders sentiment to the word/words describing the material in question. If the examined text reports the beliefs of another person, the sentence being examined is considered neutral. To identify such cases, we use the respective lists given as input to the algorithm. If the sentence contains a verb also found in the assertions or counterfactuals lists, the process is stopped, the sentence is appointed with a neutral value and the analysis continues for the next sentence. In the case of the existence of verbs denoting counterfactual, the list is employed taking into consideration that we contemplate at segregating secondary if-clauses that are dependent from a question verb, that is when the main clause they depend on regards indirect speech, versus if-clauses that don't depend on question verbs, that is when the main clause they depend on regards counterfactuals [21].

Otherwise, we investigate the existence of valence shifters within the examined sentence. At this moment, we take into account negations and comparisons. In the first case, if a word that discloses negation is present (e.g. "no", "not", etc.) and it is syntactically associated with the found affective term, the latter term is considered to carry the opposite polarity value. In the case of comparisons, the affective term pertains to both the compared entities. We distinguish two general cases:

- One entity accepts the actual valence of the affective word and the other one the opposite. In specific, the valence to be accredited is decided in reference with the syntactic relationship between the keyword term under examination and the word that discloses comparison (e.g. "than").
- Both entities accept the valence of the affective word. Specially, the valence to be accredited is decided in reference with the syntactic relationship between the keyword term under examination and the word that discloses comparison (e.g. "as").

We first eliminate stop words from the keyword list. Therefore, a new set of keywords is created. For every word of the text to be processed, we examine whether it is mapped on this new set and on the other four lists. For every sentence of the input text, if an entry of the lists in the reporting verbs is met and the dependency that binds it to the text's opinion holder is of subject type, neutral sentiment is attributed. Else, for every word of the text to be processed, we examine whether it is mapped on either of the two lists containing the affective terms. If they match an entry in either of them, if no negation or comparison dependencies are met, a corresponding value gets incremented by 1, else, the reverse one.

After having traversed the whole text, we compare the two sums and the one with the highest value is considered to be the dominant one and the respective valence is attributed to the input text. If they are equal or no sentiment is detected, the text is considered to carry neutral sentiment.

4   Results

A set of experiments have taken place so as to evaluate the performance of each algorithm. At this moment, we have completed and present here the results for the first of the presented algorithms.
Given the fact that our task is a classification one, standard classification metrics from the literature have been used. In specific, we try to detect the precision, recall and accuracy values obtained from the above described input data sets.

We wanted to detect opinion in three classes, i.e., positive, negative and neutral. So, precision will show us for each of the positive class how many of the positive instances found are indeed positive; recall will show how many of the positive instances have been found out of the total number of the positive instances that should have been found are indeed positive. The same measures will be given for the other two classes; finally, accuracy will show for each data set how many instances were correctly classified as far as all three classes are concerned.

For an initial corpus of user generated reviews, we used content from the Merlot[1] repository. Merlot is an online repository distributing free access to resources for learning and online teaching. It provides learning material of higher education aiming at promoting access to scientific data and as a result to their manipulation and exploitation by research communities. Reaching its instructional objectives necessitates ensuring that the quality of its content is of high standards. It, therefore, accredits reviews and peer reviews, attending on continuously enhancing their quantity and quality. Our system aims at enabling this procedure by proposing a way of evaluating automatically opinions expressed for the learning materials and thus contributing to enabling the community accessing valuable data and promoting its scientific goals.

---

[1] www.merlot.org/

Within Merlot, we are interested in the user comments and the expert reviews associated with each educational resource. To be more specific, users and community experts have expressed their opinion in respect of its quality, its orientation and the degree to which it complies with helping the user exploit its potentials. We refer to the former category as "user comments" and to the latter as "expert reviews". The expert reviews provide an evaluation for three distinct subcategories, namely (a) Content quality, (b) potential effectiveness as a teaching tool and (c) ease of use for both students and faculty.

For each category of the corpus we have performed two experiments, as provided by the two sets of lists respectively. Our first category regards the processing of the 6792 user comments stored in the Merlot repository. These comments have been considered as attributing positive opinion with respect to a research material if they have been rated with 5 or 4 stars, neutral if they have been attributed 3 and otherwise negative.

As peer reviewers state their opinions with respect to strengths and concerns in each of the afore-mentioned subcategories, the neutral class is empty in this context. To be more specific, for each sub-category, we have tested our system's performance again with both sets of lists in the 626 texts carrying sentiment.

4.1 Construction of the Lists of affective Terms

For the experiments conducted thus far, two sets of lists from the literature have been tested as input, both of which contain positive and negative terms. No list of neutral terms has been taken into consideration, since literature doesn't provide such lists.

The first set of lists is provided by [22] and we will refer to it as the "ANEW" subset. The second is derived from SentiWordNet [23].

Namely, SentiWordNet is a lexical resource for opinion mining. It assigns to each synset (synonym set) of WordNet three sentiment scores, each representing respectively: positivity, negativity, objectivity. In specific, according to WordNet, a synset or synonym set is defined as a set of one or more synonyms that are interchangeable in some context without changing the truth value of the proposition in which they are embedded.

The values of positivity, negativity and objectivity follow the rule:

$$positivity + negativity + objectivity = 1 \quad (6)$$

where,:

- positivity describes how positive the terms contained in the synset are,
- negativity describes how negative the terms contained in the synset are and
- objectivity describes how neutral the terms contained in the synset are.

Our goal was to extract two lists of words, positives and negatives. Due to the fact that both positivity and negativity values are assigned to a word we needed to make sure the word was clearly biased. So, each of the three classes can take values from 0 to 1 and they are complementary, as deduced by the formula.

The rule we've used for extracting the lists is:

$$\frac{positivity}{negativity} \geq 0.7 \ \&\& \ objectivity \geq 0.2 \quad (7)$$

This way we check that the word has a positivity or negativity value above 70% and from the rest of the percentage, at least 20% goes to objectivity leaving only 10% max for the opposite sentiment.

By applying the check defined in (7) we make sure there is a clear bias towards the positivity or negativity and the rest is assigned to objectivity.

Finally, we have created a subset of lists from the two above mentioned subsets, i.e. the ANEW and the SentiWordNet ones. To be more specific, two hash tables have been created one containing the positive ANEW terms and the other the positive SentiWordNet terms. If a key of the first table wasn't also a key entry in the second one, it was added in the new list. Having applied the same de-duplication procedure for the negative terms, we obtained two new lists, containing all terms of the first and the second list with unique entries.

4.2 Results for the Affective Term Frequency Algorithm

The respective results of each subcategory are presented in the following tables.

Tables 1 and 2 show the precision and recall achieved by the current system version for user comments and experts reviews respectively. What is of interest is that the User Comments present very high accuracy in the positive class, unlike the negative one. The reason for such results is the unbalanced distribution of instances per class in the specific input set. Moreover, we can tell that, when prior sentiment knowledge is received as input via the ANEW lists, precision and mostly recall is higher than when SentiWordNet or Mixed lists are adopted.

Table 1. Precision and Recall for User Comments

List	Positive		Negative		Neutral	
	Precision	Recall	Precision	Recall	Precision	Recall
ANEW	0.999	0.823	0.0	0.0	0.031	1.0
SentiWN	0.995	0.390	0.0	0.0	0.031	1.0
Both	0.996	0.740	0.0	0.0	0.010	0.242

Table 2. Precision and Recall for Expert Reviews

Subcategory	List	Positive		Negative	
		Precision	Recall	Precision	Recall
Content Quality	ANEW	0.737	0.940	0.930	0.353
	SentiWN	0.660	0.310	0.392	0.170
	Both	0.793	0.852	0.650	0.314
Effectiveness	ANEW	0.710	0.900	0.860	0.400
	SentiWN	0.704	0.458	0.707	0.220
	Both	0.721	0.853	0.643	0.371
Ease of Use	ANEW	0.860	0.844	0.864	0.270
	SentiWN	0.565	0.240	0.350	0.200
	Both	0.740	0.700	0.591	0.260

Table 3 shows the overall accuracy of the module as presented in every subcategory. Here, again, we can notice the higher values obtain by ANEW lists, followed by the Mixed ones. Furthermore, it's worth mentioning that precision and recall figures of the positive and negative classes are fairly higher than the accuracy of the overall system. The reason for such a difference lies in the fact that our input didn't include neutral class instances.

Table 3. Accuracy achieved

Input Type	Lists	Accuracy
User Comments	ANEW	0.823
	SentiWN	0.390
	Both	0.734
Expert Reviews	Content Quality	ANEW	0.550
		SentiWN	0.300
		Both	0.580
	Effectiveness	ANEW	0.534
		SentiWN	0.390
		Both	0.552
	Ease	ANEW	0.530
		SentiWN	0.262
		Both	0.471

5   Conclusions & Future Work

The preliminary results of the sentiment analysis on user comments in the context of a repository of educational resources indicated that there can be valuable qualitative information that can be added to a recommendation service and be used to adjust the perceived "rating" of a given resource by a specific user. The accuracy of the first of the examined algorithms, while satisfactory, leaves room for improvement. We expect that more elaborate techniques that introduce association of entities and contextual information will produce better results. However, it is important to note that sentiment analysis does not suffer much from domain differentiation or variability on user roles (that is, the results for expert reviews and general user comments presented similar success). An interesting remark regarding the linguistic characteristics of the examined content is that the criticism is usually employed using mild terminology, which is in contrast of user-generated reviews for products/ movies etc. This indicated the necessity of repeating the experiments with different thresholds for the restriction employed in (7), as a review considered neutral or even positive by the system is actually negative but the phrasing of the reviewer is not strong enough to provide strong indications of his/ her polarity.

Our immediate next step is to measure the performance of the remaining sentiment analysis algorithms and draw conclusions for their suitability in the context of large-scale educational repositories. Following the finalization of the sentiment analysis methodology, we intend to incorporate the results in the recommendation system for the Organic.Edunet platform in order to produce a "suitability score" for a user-resource pair or a community-resource pair. Our aim is to define this score in a way that reflects both quantitative (visits, access time, downloads) and qualitative (opinions) characteristics. The foundation of our envisioned approach is the building of a connectivity graph between the system's users and communities with respect to their profile similarity and their interests as perceived by their activity. The sentiment analysis module will be used for extracting their opinion on the overall quality of the resources they have reviewed or commented on, as well as more specific characteristics (ease of understanding, innovation) where such features can be recognized by the linguistic analysis of the reviews/ comments. The sentiment score will be incorporated in the calculation of the trust and reputation scores of the users and resources will be proposed to other members of the community based on these scores.

6   Acknowledgments

7   References

1. B. Sarwar, G. Karypis, J. Konstan and J. Riedl: Analysis of recommendation algorithms for e-commerce. In proceedings of the 2nd ACM Conference on Electronic Commerce, pp. 158-176, Minneapolis, MN, October 2000
2. N. Manouselis, R. Vuorikari and F. Van Assche: Collaborative Recommendation of e-Learning Resources: An Experimental Investigation, Journal of Computer Assisted Learning, 26(4), 227-242, 2010
3. N. Manouselis, H. Drachsler, R. Vuorikari, H. Hummel and R. Koper: "Recommender Systems in Technology Enhanced Learning", in Kantor P, Ricci F, Rokach L, Shapira B (eds), Recommender Systems Handbook, pp. 387-415, Springer US, 2011
4. J. L. Herlocker, J. Konstan, L.G. Terveen and J. Riedl: Evaluating Collaborative Filtering Recommender Systems. ACM Trans. Inf. Syst., Vol. 22(1), pp. 5-53, January 2004
5. M.-A. Sicilia, E. Garca-Barriocanal, S. Sanchez-Alonso and C. Cechinel: "Exploring user-based recommender results in large learning object repositories: the case of MERLOT", in Manouselis N, Drachsler H, Verbert K, Santos OC (eds) Proceedings of the 1st Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2010), Procedia Computer Science, 1(2):2859-2864, 2010.
6. B. Pang and L. Lee: "Opinion Mining and Sentiment Analysis", Now Publishers Inc., 2008

7. K. Durant, and M. Smith: Mining Sentiment Classification from Political Web Logs.In: Proceedings of Workshop on Web Mining and Web Usage Analysis of the 12th ACM SIGKDD International Conference on Knowledge Kennedy, A., Inkpen, D.: Sentiment Classification of Movie and Product Reviews Using Contextual Valence Shifters. Computational Intelligence, 110–125 (2006)

8. P. Turney: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In: Proceedings of ACL, Philadelphia, PA, July 2002, pp. 417–424 (2002)

9. J. Kamps, M. Marx and R.J. Mokken: Using WordNet to Measure Semantic Orientation of Adjectives. In: LREC 2004, vol. IV, pp. 1115–1118 (2004)

10. V. Hatzivassiloglou and J. Wiebe: Effects of Adjective Orientation and Gradability on Sentence Subjectivity. In: Proceedings of the 18th International Conference on Computational Linguistics, New Brunswick, NJ (2000)

11. A. Kennedy and D. Inkpen: Sentiment Classification of Movie and Product Reviews Using Contextual Valence Shifters. Computational Intelligence, 110–125 (2006)

12. A. Andreevskaia, S. Bergler and M. Urseanu: All Blogs Are Not Made Equal: Exploring Genre Differences in Sentiment Tagging of Blogs. In: International Conference on Weblogs and Social Media (ICWSM-2007), Boulder, CO (2007)

13. J. Akshay: A Framework for Modeling Influence, Opinions and Structure in Social Media. In: Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, Vancouver, BC, July 2007, pp. 1933–1934 (2007)

14. M. Hu and B. Liu, "Mining and Summarizing Customer Reviews", in Proc. of the 10th International Conference on Knowledge Discovery and Data Mining (KDD 2004), pp. 168-177, Seattle, USA, 22–25, August 2004

15. R. Feldman, B. Rosenfeld, R. Bar-Haim and M. Fresko, "The Stock Sonar — Sentiment Analysis of Stocks Based on a Hybrid Approach", in Proc. of the 23rd Innovative Applications of Artificial Intelligence Conference (IAAI 2008), pp. Chicago, USA, 13–17, July 2008

16. S. Kim and E. Hovy, "Determining the Sentiment of Opinions", in Proc. of the 20th International Conference on Computational Linguistics (COLING 2004), Article No.: 1367 pp. http://dl.acm.org/citation.cfm?id=1220555, Geneva, Switzerland, 23-27, August 2004

17. N. Godbole, M. Srinivasaiah and S. Skiena, "LargeScale Sentiment Analysis for News and s", in Proc. of the 1st International Conference on Wes and Social Media (ICWSM 2007), pp. short paper Boulder, USA, 26-28, March 2007

18. J. Kim, J. Li and J. Lee, "Discovering the Discriminative Views: Measuring TermWeights for Sentiment Analysis", in Proc. of the 47th Annual Meeting of the Joint conference of the Association for Computational Linguistics (ACL 2009) and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (IJCNLP 2009), pp. 253–261, Suntec, Singapore, 2-7, August 2009

19. X. Ding, B. Liu and P. Yu, "A Holistic Lexicon-Based Approach to Opinion Mining", in Proc. of the 1rst International Conference on Web Search and Web Data Mining (WSDM 2008), pp. 231-240, Palo Alto, USA, 11-12, February 2008

20. T. Nasukawa and J. Yi, "Sentiment Analysis: Capturing Favorability Using Natural Language Processing", in Proc. of the 2nd International Conference on Knowledge Capture (K-CAP 2003) held in conjunction with the 2nd International Semantic Web Conference (ISWC 2003), pp. Sanibel Island, USA, 23–25, October 2003

21. R. Narayanan, B. Liu, and A. Choudhary, "Sentiment Analysis of Conditional Sentences", in Proc. of the 6th Conference on Empirical Methods in Natural Language Processing (EMNLP 2009), pp. 180–189, Singapore, Singapore, 6-7, August 2009

22. M. Bradley and P. Lang, "Affective norms for English words (ANEW): Instruction manual and affective ratings", Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, 1999

23. B. Ohana, B. Tierney, "Sentiment Classification of Reviews Using SentiWordNet", in Proc. of the 9th Information Technology and Telecommunication Conference (IT&T 2009), pp. Dublin, Ireland, 22 - 23, October 2009

Affective Issues in Semantic Educational Recommender Systems

Olga C. Santos and Jesus G. Boticario

aDeNu Research Group. Artificial Intelligence Dept. Computer Science School. UNED
C/Juan del Rosal, 16. Madrid 28040. Spain
ocsantos@dia.uned.es, jgb@dia.uned.es
http://adenu.ia.uned.es

Abstract. Addressing affective issues in the recommendation process has shown their ability to increase the performance of recommender systems in non-educational scenarios. In turn, affective states have been considered for many years in developing intelligent tutoring systems. Currently, there are some works that combine both research lines. In this paper we discuss the benefits of considering affective issues in educational recommender systems and describe the extension of the Semantic Educational Recommender Systems (SERS) approach, which is characterized by its interoperability with e-learning services, to deal with learners' affective traits in educational scenarios.

Keywords: Educational Recommender Systems, Affective computing, Emotions, Technology enhanced learning, E-learning services.

1 Introduction

Affective issues have been considered to personalize the system response taking into account the corresponding affective states modelled. Two competing approaches exist to study the affect: 1) the categorical representation of discrete states in terms of a universal emotions model assuming that affective experiences can be consistently described by unique terms between and within individuals, and 2) the dimensional representation of affective experiences which assumes that the affect can be broken down into a set of dimensions. As to the former, several authors have proposed their own set of universal emotions, being probably Ekman's work the most popular [7]. Regarding the latter, the dimensional model was introduced by Mehrabian [14] as the pleasure-arousal-dominance space, which describes each emotive state as a point in a three-dimensional space. The pleasure dimension has been referred to as valence by many authors and the dominance dimension is often not considered. In any case, valence accounts for the pleasantness of the emotion, arousal for the strength of the emotion and dominance describes whether the user is in control of her emotions or not.

From the educational point of view, there is agreement in the literature that affect influences learning (e.g. refer to the references compiled in [17, 2, 25]). Many research works on user's affective state in education have been carried out in the field

of intelligent tutoring systems [5, 23, 19]. Moreover, from the recommender systems field, several experiments have shown some improvements when considering affective issues in the recommendation process [11, 1, 25, 18, 26].

In this paper we discuss, from the modelling viewpoint, how to deal with affective issues in the recommendation process in educational scenarios from a generic and interoperable perspective by extending the approach of Semantic Educational Recommender Systems (SERS) to deal with the emotional state of the learner.

The paper is structured as follows. First, we present related research, commenting on how affective issues are managed, introducing how emotions are considered in recommender systems and finally, reporting examples of recommender systems that deal with affective issue in educational scenarios. Then, we introduce the SERS approach and its modelling issues, highlighting its interoperability features with existing e-learning services. After that, we describe how the SERS modelling approach can be extended to deal with affective issues. Finally, we comment on the ongoing works.

2 Related research

Affective modelling [4] is a sub-area of affective computing [16] that involves i) detection of users' emotion and ii) adaptation of the system response to the users' emotional state. Aesthetic emotional responses (i.e. those produced by investigating the intrinsic emotions contained in the observed elements) can be either collected 1) *directly through questionnaires* such as the Self Assessment Manikin - SAM [3] which follows the dimensional model of emotions, or 2) *inferred* through data gathered from the analysis of i) *physiological sensors to detect internal changes* [15], ii) eye positions and eye movement measures with an *eye tracker* [6]; and iii) *observation* of user physical actions in an unobtrusively manner, such as from a) *keyboard and mouse* interactions [8]; b) *facial and vocal* spontaneous expressions [28] or c) *gestures* [12]. Combinations of multiple sources of data and contextual information have improved the performance of affect recognition [28].

The idea behind considering affective issues in educational recommender systems is that emotional feedback can be used to improve learning experiences [25]. Two strategies can be carried out related to emotions feedback [2]: 1) emotional induction, when promoting positive emotions while engaged in a learning activity, and 2) emotional suppression, when the focus on an existing emotion disrupts the learning process. Anyway, it is difficult to determine how best to respond to an individual's affective state [19], so there are open issues to be investigated, such as "at which emotion state will the learners need help from tutors and systems" [25]. To answer this question, observational techniques on tutoring actions can be carried out to facilitate the externalization of the tutors' decision-making processes during the tutoring support [17].

Moreover, students' personality characteristics can also impact on how students respond to attempts to provide affective scaffolding [19] and accounts for the individual differences of emotions in motivation and decision making [27]. Personality is commonly measured with the Five Factor Model - FFM [9].

In this context, to date there have been a few recommender systems in educational scenarios that have considered affective issues. For instance to better recommend courses according to the inferred emotional information about the user [10] or to customize delivered learning materials depending on the learner emotional state and other issues from the learning context [25]. These systems are typical applications of recommender systems in the educational domain, which mainly focus on recommending courses or learning objects [13, 22].

Last but not least, note that as for interoperability issues are concerned, although most recommenders are stand-alone applications, efforts are recently being made to integrate affective recommendation support with existing e-learning services, like the SAERS approach (introduced in the next section) or the Learning Resources Affective Recommender (LRAR) widget[1]. This widget aims to provide the list of most suitable resources given the affective state of the learner, provided that the learner fills in i) her current affective state (flow, frustrated, etc.) and ii) her learning objectives.

In summary, works in several related fields suggest that educational recommender systems can benefit from managing learners' affective states in the recommendation process. A key research question is how educational recommender systems can model the affective issues involved during the learning process to be able to properly detect them and provide appropriate recommendations to learners. For this, the involvement of educators has been suggested. Moreover, to take advantage of existing technological infrastructures in current educational scenarios, interoperability with external components should be achieved.

3 Semantic Affective Educational Recommender Systems

In this section we present the modelling issues involved in developing Semantic Affective Educational Recommender Systems (SAERS), which consider affective issues in the so called SERS (i.e. Semantic Educational Recommender Systems) approach [20]. As in the SERS approach, this extension takes advantage of existing standards and specifications to facilitate interoperability with external components.

3.1 The SERS approach

The SERS approach [20] enriches the recommendation opportunities of educational recommender systems, going beyond aforementioned typical course or contents recommendations. It has been proposed to extend existing e-learning services with adaptive navigation support, where both passive (e.g. reading) and active (e.g. contributing) actions on any e-learning system object (e.g. content, forum message, calendar event, blog post, etc.) can be recommended to improve the learning performance in terms of learning efficiency (use less amount of learning resources to achieve the learning goals), learning effectiveness (more learning activities done and more learners achieving the learning goals), satisfaction (better perception of the course experience), course engagement (more continuous and frequent accesses to the

---

[1] http://www.role-widgetstore.eu/specification/learning-resources-affective-recommender

course) and knowledge acquisition (better scoring in the course evaluation). Here recommendations are offered as a list of links of suggested actions, which provide access to explanations and feedback on demand [20].

This adaptive navigation support can be offered in terms of a service oriented architecture that provides interoperability with the different components involved: 1) **e-learning service** -initially applied to learning management systems, but extensible to personal learning environments- where the learner carries out the educational tasks, 2) **user model**, which characterizes the learner needs, interests, preferences, etc., 3) **device model**, which stores the capabilities of the device used by the learner to access the course space, 4) **SERS admin**, which supports the recommendations design, and 5) **SERS server**, which is the reasoning component. The goal of the SERS admin is to support the recommendations design process in two complementary ways: i) involving educators in the recommendations elicitation process with the user-centred design methodology called TORMES (Tutor Oriented Recommendations Modelling for Educational Systems) [21] and ii) applying recommendation algorithms. In turn, SERS server consists in a knowledge-based recommender that store rules, which are managed according to their applicability conditions in order to recommend appropriate actions to be carried out for the current learner (with her individual features, preferences, etc.) in her current context (including course activity, course history, device used, etc.). The information that is modelled and managed among the different components can be described in terms of available standards and specifications (e.g. IMS, W3C, ISO), as discussed elsewhere [20].

With respect to modelling these recommendations, they are described in terms of a recommendations model which semantically characterizes the recommendations in order to bridge the gap between their description by the educator and the recommender logic when delivering recommendations in the running course. The recommendation model consists of the following 5 elements:

- **type:** specifies *what* to recommend, that is, the action to be done on the object of the e-learning service. For instance, post a forum message.
- **content**: defines *how* to convey the recommendation, in terms of the textual information presented to inform the learner about the recommendation.
- **runtime information**: describes *when* to produce the recommendation, which depends on defining the learner features, device capabilities and course context that trigger the recommendation.
- **justification**: informs *why* a recommendation has been produced, providing the educational rationale behind the action suggested.
- **recommendation features**: additional semantic information that compiles features *which* characterize the recommendations themselves, such as i) their classification into a certain category from a predefined vocabulary, ii) their relevance (i.e. a rating value for priorization purposes), iii) their appropriateness for a certain part of the course, and iv) their origin, that is, the source that originated the recommendation (e.g. proposed in the course design, defined by the tutor during the course run, popular among similar users, based on user preferences).

Details about the SERS approach and the recommendations model can be read elsewhere [20]. Next, we comment how the SERS approach can be extended to model affective issues in an interoperable way.

3.2 From SERS to SAERS

In this section, we present how to consider affective issues in the SERS approach, assuming also a multimodal enriched environment where sensors (obtain data from the users in the environment) and actuators (produce data to the users in the environment) interact with the learners. Correspondingly, it is named SAERS (Semantic Affective Educational Recommender System). This extension involves modelling and interoperability issues: 1) user centred design of the recommendations, 2) enrichment of the recommendation model and 3) definition of new services in the architecture.

3.2.1 User centred design of the recommendations

From Section 2, dealing with affective information in educational recommender systems is an open issue. Some authors (see [17]) have proposed applying observational techniques on tutoring actions to facilitate the externalization of the tutors' decision-making processes during the tutoring support in order to find out how and when to respond to the learners' affective states.

Following that approach, TORMES methodology can be used to involve educators in identifying when, what and how the emotional feedback needs to be provided to each particular learner in each educational scenario. In particular, TORMES adapts the ISO standard 9241-210 to guide educators in eliciting and describing recommendations with educational value for their scenarios [21]. The application of TORMES involves several educators in the process, so it is costly in terms of resources. However, in our view, this is the most informative way to get the knowledge needed to be able to properly take into account affective issues in educational recommendations. This approach pays off since the recommendations can be provided and adapted to different courses and situations, and eventually are managed by the recommender, which takes into account the learner evolving process. When a large sample of educational affective recommendations generated with TORMES is available, the research question should move from identifying recommendation opportunities that deal with affective issues to finding appropriate algorithms that design affective recommendations with or without the involvement of educators.

TORMES methodology can be carried out at any time in the course life cycle. However, if the course has not been run yet, the input data would come from similar past courses and the associated educational experience in them. Four activities are defined: 1) understanding and specifying the context of use, 2) specifying the user requirements, 3) producing design solutions to meet user requirements, and 4) evaluating designs against requirements. In each of these activities, relevant information to consider the affective issues in the recommendations process during the course execution can be gathered, as follows:

- **Context of use.** The goal of this activity is to identify the context of use where the recommendations are to be delivered. Information can be gathered from two sources. On the one hand, individual interviews to educators that can serve to elicit best practices from their educational experiences. Here, the interviewer should ask

the educator if she takes into account the emotional state of their learners, and if so, what features she takes into account to detect the learners' affective state (educator detection approach) and how she reacts to it by describing the emotional feedback provided (educator adaptation approach). On the other hand, data mining analysis can be done on data gathered from learners interactions in the course to complement the initial description of the context of use obtained from the interviews, mainly adding precision (e.g. from the interview, the educator can mention the she thinks that learners with very infrequent contributions in the course space are low motivated, and the data mining techniques can be used to cluster learners in several groups regarding their engagement in the course and their motivation level in order to identify the particularities of low engaged learners with low motivation). To extract relevant information regarding affective issues, the data mined should include, if available, i) the answers given by the learners to specific questionnaires such as the SAM to compute the emotions along predefined dimensions and the FFM to obtain the learners' personality traits, ii) the data gathered by physiological sensors and eye-trackers, and iii) from non-obtrusive observations such as keyboard and mouse interactions, facial and vocal spontaneous expressions and gestures.

- **Requirements specification.** Following the scenario based approach [29] that proposes the definition of a problem and its counterpart solution scenario, the information obtained from the activity 'Context of use' is used to build representative scenarios of the tutoring task in order to identify recommendation opportunities in them, where the problem scenario identifies the situations where learners lack of support and the solution scenario avoids or minimizes those problematic situations by offering appropriate recommendations. The goal is to extract knowledge from the educators on what the requirements are for the recommendations within the given context of use and identify an initial set of recommendations. The information mined in the previous activity can be used here to propose specific values for the applicability conditions of the recommendations proposed. For instance, following the above example, if most low engaged and low motivated learners are characterized as solitary in the extraversion trait of the FFM and they have entered in the course no more than 12 times, these quantitative information can be used by the educator to fill in the corresponding applicability conditions (e.g., the recommendation is to be delivered to learners with the following values in their user model: *extraversion = solitary* and *number_of_sessions_in_course < 12*). As a result, an initial version of each of the recommendations proposed is described in terms of the recommendations model. The affective issues are to be included in this description. Hence, the recommendation model needs to be enriched with this information (see Section 3.2.2).

- **Create design solutions.** The goal of this activity is to validate the recommendations proposed in the previous activity by a group of experienced educators. Specifically focus groups are used to involve several educators in validating the initial set of recommendations elicited from the scenarios in the previous activity in order to revise the recommendations obtained in the solution scenario and come to an agreement. Educators involved in the validation should

have experience with affective computing to be able to validate the recommendations from that perspective.

- **Evaluation of designs against requirements.** In this activity, affective designed recommendations can be delivered in the e-learning system and allow educators and learners to evaluate them in their context by rating their relevance and classifying them in terms of their conceptual model. Preferably, the running prototype can be a functional system, but if that is not possible, a Wizard of Oz can be used to simulate the response of the system.

In this way, TORMES helps educators to understand the recommendation needs in their scenarios and supports them in eliciting sound recommendations that address cognitive, meta-cognitive, social and affective issues required when learners interact with their courses online. Moreover, TORMES also supports the changing of educational needs since the process is iterative and new recommendations can be added at any time during the course execution. Eventually, a set of semantically described oriented recommendations are ready to be automatically delivered to learners following a rule-based approach.

3.2.2 Enrichment of the recommendation model

As anticipated during the description of the activity 'Requirements specification' in the previous section, the SERS recommendations model needs to be extended to be able to describe the affective recommendations elicited with TORMES. In particular, up to now, we have detected the need to extend three elements of the recommendations model to include the modelling of affective issues.

The **content** element defines how to convey the recommendation to the learner. In the SERS approach, the recommendations are offered as a list of links of suggested actions. Therefore, the information to provide is the text to be shown to the learner in the recommendation areas of the course space. However, in a multimodal enriched environment, recommendations can be delivered to the learners in different ways. Therefore, this element needs to be extended with an attribute that describes the *modality* in which the recommendation has to be delivered to the learner, for instance, text or voice. Moreover, the actuators can produce the recommendations to the learner in different ways, and these ways can depend on the emotions handled [30]. For instance, a recommendation to be delivered by voice can be done with a calm tone or with an angry tone. Thus, another attribute needs to describe the ***emotional delivery state***.

The **runtime information** element that describes the applicability conditions that trigger the recommendations has to consider also the ***user personality*** (e.g. to describe the extraversion trait of the FFM) and ***emotional states*** as attributes that describe the user features to be taken into account.

The **justification** element provides the educational rationale behind the action suggested, so the affective issues considered should explicitly be mentioned in the justification text. A new attribute with this information can be added (e.g. ***affective support***).

3.2.3 New services in the architecture

To cope with the aforementioned modelling issues, from the architectural point of view, new services need to be added to the original service oriented architecture. The purpose here is to support new functionalities to cover the detection of emotions and the provision of emotional feedback in a multimodal environment. These services are: 1) *emotional data processing*, which collects the input from the different sources of emotional data available, 2) *multimodal emotions detection*, which combines the different sources of emotional data gathered to recognize the emotional state of the learner, and 3) *emotions delivery*, which delivers the recommendation to the learner in the corresponding affective modality. These services can be provided by the corresponding components, as shown in Figure 1. The sense of the arrows indicates the initiator of the information flow (request or sending).
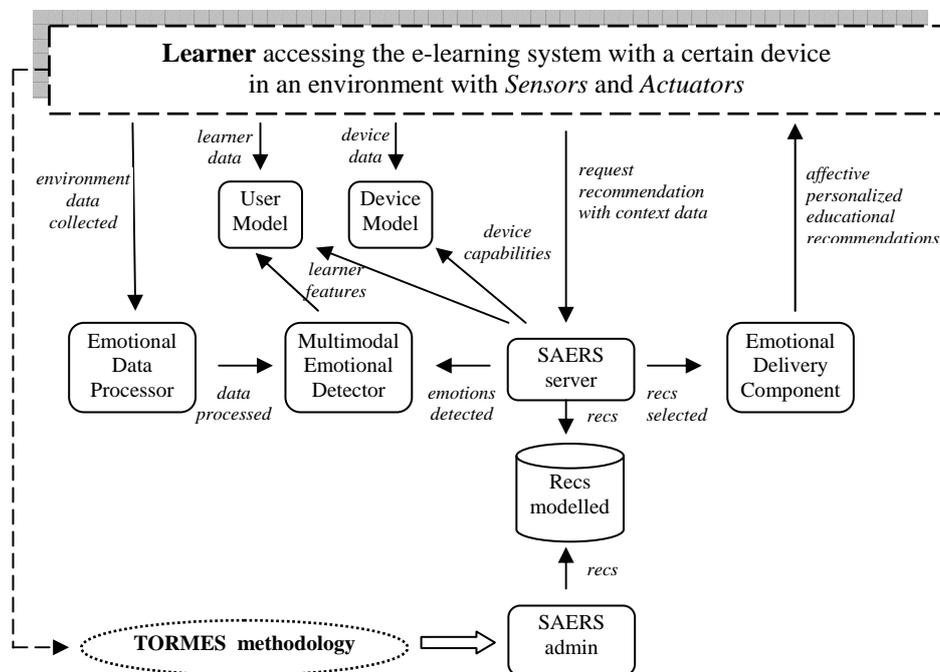

Figure 1. Components and data flow in the SAERS approach

The figure shows that the learner can be placed in a rich environment where sensors (defined in a general term) get data from her and actuators provide data to her at the same time that she is taking a course in an e-learning system through a certain device (e.g. PC, laptop, mobile, etc.) which might be combined with assistive technology (e.g. Braille line, speech recognition software, screen magnifier, among others) if the user requires some accessibility support.

At certain point during the learning process, a recommendation request is received by the SAERS server for a specific learner with details about her context in the learning environment and the device used to access. As in the SERS approach, the

SAERS server request data about the user and the capabilities of the device to the corresponding User Model and Device Model. Now, the SAERS needs additional information about the emotional state of the user, which can be requested to the Multimodal Emotional Detector. This component computes the affective state of the learner from the data received by the Emotional Data Processor as well as the information about the learner's personality stored in the User Model. The data gathered from the environment's sensors by the Emotional Data Processor consists in physiological data, eye positions and movements and physical interactions of the user (movements of the mouse, uses of the keyboard, voice or gestures). As a result, the Multimodal Emotional Detector can recognize the emotional state of the current learner and pass it to the reasoning component (SAERS server) so it can select the appropriate recommendations taking into account the current affective state of the learner.

Therefore, with that information, the SAERS server looks for exiting recommendations whose applicability conditions matches the user features and emotions, the device capabilities and the educational context. These recommendations have been designed and properly modelled through the SAERS admin with TORMES methodology. The resulting selected recommendations that are instantiated for the given request are passed to the Emotional Delivery Component, which adds the corresponding affective state to the response sent back to the environment, so the actuator selected can deliver the personalized educational oriented recommendations to the learner with the appropriate affective state.

As described in [20], the information exchanged by the different components involved in the SERS approach follows existing standards and specifications from IMS, ISO and W3C. To deal with the emotional information, the Emotion Markup Language (EmotionML) [24] proposed by the W3C to allow a technological component to represent and process data, and to enable interoperability between different technological components processing the data can be used. W3C EmotionML is conceived for 1) manual annotation of data such as videos, of speech recordings, of faces, of texts, etc., 2) automatic recognition of emotion-related states from user behaviour including information from physiological sensors, speech recordings, facial expressions, etc., as well as from multi-modal combinations of sensors, and 3) generation of emotion-related system behaviour providing responses, which may involve reasoning about the emotional implications of events, emotional prosody in synthetic speech, facial expressions and gestures of embodied agents or robots, the choice of music and colours of lighting in a room, etc.

4 Ongoing works

In order to evaluate our approach we are running several experiments in the context of the MAMIPEC project (Multimodal approaches for Affective Modelling in Inclusive Personalized Educational scenarios in intelligent Contexts - TIN2011-29221-C03-01). Our goal is twofold. On one hand, detect emotions from users' interactions in the e-learning environment through multiple sources (i.e. questionnaires and sensors). On the other hand, use that information to elicit appropriate recommendations with

Olga C. Santos and Jesus G. Boticario

TORMES methodology that take into account the emotional needs of the learners, and deliver affective educational oriented recommendations personalized to the learner through the e-learning environment by the extended SERS approach, that is, the SAERS.

Up to now, we have carried out a pilot with two users to test the appropriateness of the activities designed to induce emotions while the learner is taking the course activities. Participants were asked to perform mathematical exercises with several levels of difficulty and varied time restrictions. At the beginning they filled in the FFM questionnaire, and after each exercise they were asked to fill in the SAM scale to measure the caused emotions with the dimensional approach. With that experiment, we aim to check if the induced emotions can be measured with the technological infrastructure that we have prepared, which combines diverse sources for gathering emotional data from users. The pilot was successful in the sense that we were able to integrate and record data from different sources simultaneously, namely, eye movements from an eye tracker, face expressions from Kinect, video from a web cam, heart and breath parameters from physiological sensors, and mouse and keyboard movements. We are currently processing the data obtained trying to automate its processing for forthcoming sessions.

The next steps consist in revising the educational scenario proposed for this pilot and applying the TORMES methodology to elicit and design affective educational oriented recommendations taking into account the extensions to the SAERS approach to deal with the modelling issues, such as the new attributes proposed for some of the elements of the recommendations model (i.e. modality, emotional delivery, user personality, emotional state, affective support). The development of the components to provide the services required (i.e. emotional data processing, multimodal emotions detection and emotions delivery) is also part of future works. The W3C EmotionML language is to be considered to facilitate the exchange of the affective information among the components of the service oriented architecture.

Acknowledgments

References

1. Arapakis, I., Moshfeghi, Y., Joho, H., Ren, R., Hannah, D., Jose, J., Gardens, L. Integrating facial expressions into user profiling for the improvement of a multimodal recommender system. In: Proceedings of the IEEE International Conference on Multimedia and Expo, p. 1440–1443, 2009.
2. Blanchard, E.G., Volfson, B., Hong, Y.J. Lajoie, S.P. Affective Artificial Intelligence in Education: From Detection to Adaptation. In Proceedings of the 2009 conference on Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling (AIED 2009), 81-88, 2009.
3. Bradley, M.M., Lang, P.J. Measuring Emotion: The Self-Assessment Manikin and the Semantic Differential. Journal of Behavior Therapy und Experimental Psychiatry, 25 (I), 49-59, 1994.
4. Carberry, S., de Rosis, F. Introduction to special issue on affective modeling and adaptation. User Model. User-Adapt. Interact. 18(1), p. 1–9, 2008.
5. Craig, S. D., Graesser, A. C., Sullins, J., Gholson, B. Affect and learning: An exploratory look into the role of affect in learning with AutoTutor. Journal of Educational Media, 29 (3), 241-250, 2004.
6. de Lemos, J., Sadeghnia,G.R., Ólafsdóttir, I., Jensen, O. Measuring emotions using eye tracking. Proceedings of Measuring Behavior, p. 226, 2008.
7. Ekman, P.: Basic emotions. In: Dalgleish, T., Power, T. (eds.) Handbook of Cognition and Emotion.Wiley, New York, 1999.
8. Filho, J., Freire, O. On the Equalization of Keystroke Time Histograms. Pattern Recognition Letters. Elsevier, Vol. 27, Issue 12, pp. 1440-1446, 2006.
9. Goldberg, L.R. The structure of phenotypic personality traits. American Psychologist 48 (1), p. 26–34, 1993.
10 .Gonzalez, G., de la Rosa, J.L., Montaner, M.., Delfin, S. Embedding Emotional Context in Recommender Systems. In Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering Workshop (ICDEW '07). p. 845-852, 2007.
11.Kuo, F.F., Chiang, M.F., Shan, M.k:, Lee, S.Y. Emotion-based music recommendation by association discovery from film music. In Proceedings of the 13th annual ACM international conference on Multimedia (MULTIMEDIA '05), 507-510, 2005.
12.Mahmoud, M., Baltrušaitis, T., Robinson, P., Riek, L. 3D corpus of spontaneous complex mental states. In Proceedings of the International Conference on Affective Computing and Intelligent Interaction (ACII 2011). Lecture Notes in Computer Science, 2011.
13.Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., Koper, R. Recommender Systems in Technology Enhanced Learning, in Kantor P., Ricci F., Rokach L., Shapira, B. (Eds.), Recommender Systems Handbook: A Complete Guide for Research Scientists & Practitioners, 2010.
14.Mehrabian, A. Pleasure-arousal-dominance: a general framework for describing and measuring individual differences in temperament. Curr. Psychol. 14(4), p. 261–292, 1996.
15.Oehme, A., Herbon, A., Kupschick, S., Zentsch, E. Physiological correlates of emotions. Workshop on Artificial Societies for Ambient Intelligence. Artificial Intelligence and Simulation of Behaviour, in association with the AISB '07, 2-4 April, 2007.
16.Picard, R.W. Affective Computing. MIT Press, Cambridge, 2000.
17.Porayska-Pomsta, K., Mavrikis, M., Pain, H. Diagnosing and acting on student affect: the tutor's perspective. User Model. User-Adapt. Interact. 18(1-2): 125-173, 2008.
18.Qing-qiang, L., Zhu, J., Kong, T. RSED: a Novel Recommendation Based on Emotion Recognition Methods. International Conference on Information Engineering and Computer Science (ICIECS 2009), 1–4, 2009.

19. Robison, J.L., McQuiggan, S.W., Lester, J.C. Developing Empirically Based Student Personality Profiles for Affective Feedback Models. Intelligent Tutoring Systems 2010, 285-295, 2010.

20. Santos O.C., Boticario J.G. Requirements for Semantic Educational Recommender Systems in Formal E-Learning Scenarios. Algorithms. 4(2), 131-154, 2011.

21. Santos, O.C., Boticario, J.G. TORMES methodology to elicit educational oriented recommendations. Lect. Notes Artif. Intell. 6738, 541–543, 2011.

22. Santos, O.C., Boticario, J.G. (Eds.) Educational Recommender Systems and Techniques: Practices and Challenges. IGI Publisher, 2012.

23. Sarrafzadeh, A., Alexander, S., Dadgostar, F., Fan, C., Bigdeli, A. How do you know that I don't understand? A look at the future of intelligent tutoring systems. Computers in Human Behavior, Volume 24, Issue 4, July 2008, 1342-1363, 2008.

24. Schröeder, M., Baggia, P., Burkhardt, F., Pelachaud, C., Peter, C., Zovato, E. Emotion Markup Language (EmotionML) 1.0. W3C Candidate Recommendation 10, 2012 May 2012

25. Shen, L., Wang, M., Shen, R. Affective e-Learning: Using "Emotional" Data to Improve Learning in Pervasive Learning Environment. Educational Technology & Society (ETS) 12(2), 176-189, 2009.

26. Tkalcic, M., Burnik, U., Kosir, A. Using affective parameters in a content-based recommender system for images. User Model. User-Adapt. Interact. 20(4), 279-311, 2010.

27. Yik, M., Russell, J.A., Ahn, C.k., Fernandez Dols, J.M., Suzuki, N. Relating the five-factor model of personality to a circumplex model of affect: a five-language study. In: McCrae, R.R., Allik, J. (eds.) The Five-Factor Model of Personality Across Cultures, Kluwer Academic Publishers, New York, 79–104, 2002.

28. Zeng, Z., Pantic, M., Roisman, G.I., Huang, T.S. A survey of affect recognition methods: audio, visual and spontaneous expressions. IEEE Trans. Pattern Anal. Mach. Intell. 31(1), 39–58, 2009.

29. Rosson, M. B. and Carroll, J. M. Usability engineering: scenario-based development of human computer interaction. Morgan Kaufmann, 2001.

30. Baldassarri, S., Cerezo, E., Seron, F.J. Maxine: A platform for embodied animated agents, Computers and Graphics, 32 (4), 430-437, 2008.

A Framework for Cross-Platform Graph-based Recommendations for TEL

Mojisola Anjorin[1], Ivan Dackiewicz[2], Alejandro Fernández[2], and Christoph Rensing[1]

[1] Multimedia Communications Lab,
Technische Universität Darmstadt, Germany
{mojisola.anjorin,christoph.rensing}@kom.tu-darmstadt.de
[2] LIFIA-UNLP, La Plata, Argentina
{idackiewicz,alejandro.fernandez}@lifia.info.unlp.edu.ar

Abstract. A Technology Enhanced Learning (TEL) ecosystem is a kind of Digital Ecosystem formed by independent platforms combined and used by learners to support their learning. We believe that recommendations made across these different platforms by exploiting the synergies between them will benefit learners. However, building such cross-platform recommender systems poses new and unique challenges for developers. In this paper, we present a framework to support the development of cross-platform recommender systems for TEL ecosystems and discuss challenges faced. The framework decouples the development of the recommender system from the evolution of the specific platforms by combining graph-based algorithms, a unified data model, and a service oriented architecture. As proof of concept, the framework was effectively applied to develop a cross-platform recommender system in a TEL ecosystem having Moodle as the Learning Management System, Mahara as the Social Networking Service and Ariadne as Learning Object Repository.

Keywords: TEL, Recommender Systems, Cross-Platform, Framework

1 Introduction

A Technology Enhanced Learning (TEL) ecosystem, is a form of a Digital Ecosystem [2] inhabited by elements from various platforms used in parallel by learners and teachers. Such a simultaneous use of platforms is often found in communities of practice [9] also known as learning networks, where learning is mostly self-directed. In this paper, we focus on a TEL ecosystem with three platforms: a Learning Management Systems (LMS), a Social Networking Service (SNS), and a Learning Object Repository (LOR). An LMS offers activities as well as discussion forums and shared spaces such as wikis. Activities rely on learning objects (LOs) such as lesson notes and presentations. The visibility of a LO is normally limited to an activity. However when an LMS is used to support self-directed learning, it becomes particularly important that learners are aware of all activities, resources and peers they could potentially gain from. Nowadays,

M. Anjorin, I. Dackiewicz, A. Fernández and C. Rensing

many learners participate in social networks connecting to other learners via Facebook[3], or posting learning tasks and following other learners on Twitter[4]. Contacts the students have on platforms such as an LMS are disconnected from the online social networks they belong to outside the classroom. It is therefore up to the students to replicate in each of these worlds the relationships they have built in the other. The potential to share knowledge and find valuable contacts across these platforms therefore remains unexploited. Initiatives such as the MIT OpenCourseWare[5] or the Ariadne Foundation[6] with its LOR demonstrate the increasing interest in collecting and sharing high quality learning material. LORs however are isolated from the LMS and SNS. There therefore exists an opportunity to provide learners with information across multiple platforms by considering the synergies between them.

In the following sections we propose a framework to empower a TEL ecosystem by generating cross-platform recommendations in each of them based on resources gained from all of them.

2 Related Work

Recommender systems based on approaches such as content based and collaborative filtering (CF) techniques have been shown to be very useful in TEL scenarios, especially in informal learning [8]. CF approaches use community data such as feedback or ratings from other users to make recommendations. Graph-based recommender techniques can be classified as neighborhood-based CF approaches [4]. A graph is used to represent the users or items as nodes and the edges as the transactions between them. PageRank [3] is an example of a graph-based approach based on a random walk similarity. Transitive associations are defined within a probabilistic framework where the similarity or affinity between nodes is calculated as a probability of reaching these nodes in a random walk on a weighted graph having a node for each state. The probability of jumping from one node to another is given by the weight of the edge connecting these nodes. In this paper, we implement PageRank using the information from the platforms that make up the ecosystem, to generate recommendations across them.

ReMashed [5] is a Mash-up Personal Learning Environment allowing learners to combine content from different Web 2.0 services to a personal view or mash-up. Learning resources are recommended using a CF approach that matches users with similar opinions and considers the learning goals of the learner. In contrast, we propose a framework to recommend activities, users and LOs across multiple platforms, thus pointing the learners to other valuable sources of information found on these different platforms without building a mash-up.

Recommender systems are often implemented as closed, internal components of larger applications having tightly coupled components. In contrast, APOS-

---

[3] http://facebook.com (last retrieved 30.06.2012)
[4] http://twitter.com (last retrieved 10.07.2012)
[5] http://ocw.mit.edu/index.htm (last retrieved 10.07.2012)
[6] http://www.ariadne-eu.org/ (last retrieved 10.07.2012)

DLE [1] for example, follows the SOA approach providing web services to publish knowledgeable person recommendations. Web services decouple the generation of recommendations from its presentation to the users. Our framework uses a similar approach. Furthermore, graph-based approaches are suitable for integrating data from various platforms, using the graph as the grounds for inter-operation. This is particularly interesting when combined with vocabularies and technologies that originate in the Semantic Web and Linked Open Data movements [6].

3 Cross-Platform Recommendation Framework

The framework is shown in Fig.1 where the TEL ecosystem comprises of an LMS, an SNS and a LOR. These platforms are independent of each other and have been implemented autonomously. The introduction of recommendations should neither increase coupling between these platforms, nor require intrusive changes that will hinder their maintenance. Moreover, the choice of platforms to be integrated must remain flexible, allowing for new alternatives to be introduced as a replacement for any of them or as a complement (i.e., there could be more than one LMS, SNS or LOR). To provide recommendations in such a TEL ecosystem, our framework adopts a service oriented architecture. The *Recommender* in Fig.1 is implemented as an independent component. It provides a parameterizable implementation of a graph-based recommender algorithm (1). The algorithm takes as input a graph with nodes representing items in each of the platforms and links representing relationships between them (2). The values given to the nodes and the weights for the edges influence how the algorithm ranks the elements. A service publishes a function that the platforms can call to retrieve recommendations (3). All changes in the platforms that are relevant to compute recommendations (i.e., to build the graph) are communicated (4) and stored by the recommender in its data model (5). The data model is also the basis for exchanging relevant data between the platforms and the recommender. Finally, there is a mapping (6) to generate the graph (i.e., the nodes and edges) from the data model. The mapping allows for the introduction of links that did not exist in the data model (e.g., links connecting semantically similar resources or links that connect users belonging to the same group).

The User Interface and Recommendation Lists: From the user's perspective, each platform introduces a recommendation list to the User Interface (UI) component. In Fig.2, the recommendation list is shown in Mahara (left side) and in Moodle (right side). The recommendations are personalized considering the user's current focus. For example, in Fig.2 recommendations are provided in Mahara for the user Albert Alonso taking into account that he is currently focused on viewing Bernard Berazategui's user profile. Depending on the recommendation strategy, the recommendation lists might include other users that Bernard has befriended, activities that he has completed, and resources that he frequently uses. Consequently, the recommendation lists contain items from any of the three integrated platforms: Activities, LOs and Users.

M. Anjorin, I. Dackiewicz, A. Fernández and C. Rensing
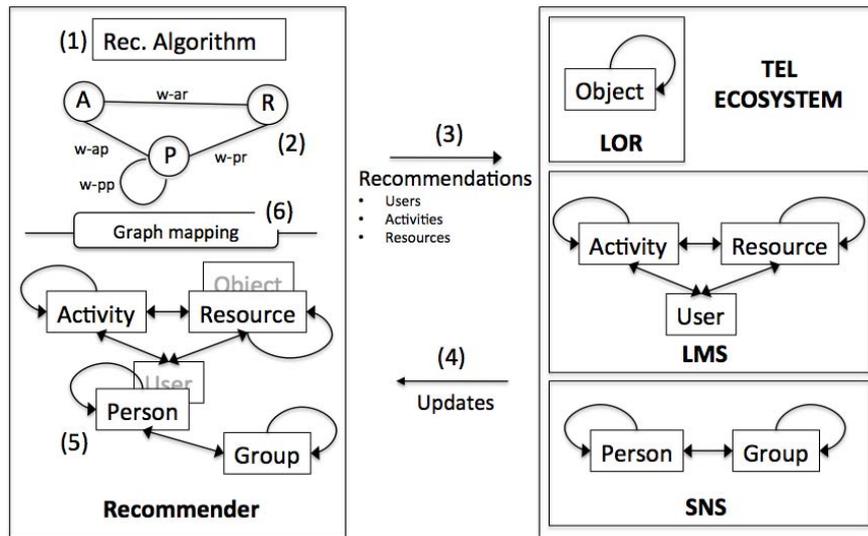
Fig. 1. Overview of the Cross-Platform Recommendations Framework

The Service Oriented Approach: The recommender component implements five core web-services. New resources are added through an addResource() service, taking as argument the unique identifier (URI) of the object. Attributes of the object and relationships are added/updated through calls to updateDataAttribute() and updateObjectAttribute() respectively. To retrieve recommendations, clients call the getRecommendations() service indicating the user and his current focus (a specific object). To encapsulate the development of the recommendations in the UI components (thus reducing coupling between these components and the rest of the functionality of the platforms) we follow a plug-in approach. Most open platforms support a plug-in extension mechanism. Our framework provides an interface that plug-ins can invoke to implement operations to display, register and handle events that correspond to changes to any of the relevant objects on the platform. Each platform is required to implement the recommender UI element as a plug-in component, ofcourse, depending on the platform, this can pose an implementation challenge.

The Data Model and Data Mapping: The data model serves two key purposes: First, it is used to create the graphs that feed the recommender algorithm. Second, it provides basic information about the objects that each of the platforms displays to the user. This approach has to remain generic enough to accommodate not only the platforms that we choose for the proof of concept (Moodle, Mahara and Ariadne) but other alternatives as well. The data model is stored in the form of triples. Each object has a unique id (a URI). Relationships between objects (objectURI, relationship, subjectURI) as well as object attributes (objectURI, attribute, value) are stored as triples. A certain object attribute relates
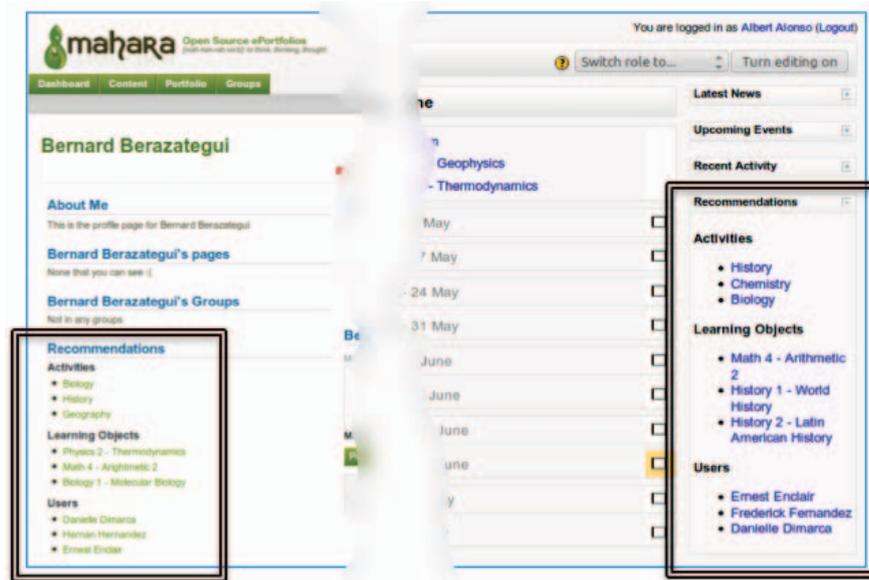
Fig. 2. Recommendation of activities, learning objects and users in Mahara and Moodle

the object to its type (e.g, a user object to the URI of the Person type). The data model aggregates information that would otherwise be disconnected, e.g, it connects LOs from the LOR to users and activities in the LMS. Therefore the definition of a common unique identifier (e.g, primary email for persons) across all platforms is needed to uniformly identify objects that are present on the different platforms, and become one in the data model. A challenge here is considering the access rights the user has in each system in order to only recommend objects the user is allowed to view. A common user authentication like single sign-on could be a solution.

The Recommender Algorithm: In this implementation, we choose the PageRank algorithm on the graph to produce a ranking of nodes. This is implemented using the JUNG (Java Universal Network/Graph) framework [7]. This ranking is the basis for the recommendation lists that are returned to clients. A graph mapping strategy generates the graph from the data model. First, it generates a node for each object (i.e., Persons, Activities and Resources). Nodes have values (e.g., the probability of reaching the node after a random jump) and the URI of the object they represent. A node's value is set in a way that increases the impact it has in the resulting ranking, e.g. the node representing the user or the object in focus starts with a higher weight. Then, the mapping strategy generates edges. The weight given to each type of edge can be configured to give certain connections higher relevance. In the current implementation, the relationships considered are user - user, user - resource, user - activity and activity - resource. The weights are calculated as the average number of relationships between the different types

of nodes i.e. the number of resources accessed by the user/ the number of resources that have been accessed by any user. In the initial experiment, about 80 relationships are considered between 12 users, 15 LOs and 6 activities.

4 Conclusion

In this paper, we propose to take advantage of the synergies that arise across multiple platforms in order to generate cross-platform recommendations in a TEL ecosystem, aiming to further enhance the learning effort of the learners. Focusing on graph-based recommendations, we discussed design and implementation challenges. Providing effective recommendations requires experimenting with different platform combinations, and graph configurations. To ease the development efforts, we propose a framework to provide recommendations in a TEL ecosystem. As a proof of concept and to demonstrate the flexibility of such a framework, an implementation was made with Moodle as LMS, Mahara as SNS and Ariadne as LOR. Future work will be to integrate additional platforms in a different constellation of a TEL ecosystem and to conduct a usability study to evaluate the recommender algorithms used in the framework.

References

1. Beham, G., Kump, B., Ley, T., Lindstaedt, S.: Recommending knowledgeable people in a work-integrated learning system. Procedia Computer Science (2010)
2. Boley, H., Chang, E.: Digital Ecosystems: Principles and Semantics. In: Inaugural IEEE Int. Conf. on Digital Ecosystems and Technologies. Cairns, Australia (2007)
3. Brin, S., Page, L.: The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems 30, 107–117 (1998)
4. Desrosiers, C., G., K.: A Comprehensive Survey of Neighborhood-Based Recommendation Methods. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. (eds.) Recommender Systems Handbook, pp. 107–144. Springer (2011)
5. Drachsler, H., Pecceu, D., Arts, T., Hutten, E., Rutledge, L., van Rosmalen, P., Hummel, H.G.K., Koper, R.: Remashed - recommendations for mash-up personal learning environments. In: Cress, U., Dimitrova, V., Specht, M. (eds.) EC-TEL. LNCS, vol. 5794, pp. 788–793. Springer (2009)
6. Fernández-Tobías, I., Cantador, I., Kaminskas, M., Ricci, F.: A generic semantic-based framework for cross-domain recommendation. In: Proc. of the 2nd Int. Workshop on Information Heterogeneity and Fusion in Recommender Systems. pp. 25–32. HetRec '11, ACM, New York, NY, USA (2011)
7. Madadhain, J., Fisher, D., Smyth, P., White, S., Boey, Y.: Analysis and visualization of network data using jung. Journal of Statistical Software 10, 1–35 (2005)
8. Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., Koper, R.: Recommender Systems in Technology Enhanced Learning. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. (eds.) Recommender Systems Handbook, pp. 387–415. Springer (2011)
9. Wenger, E.: Communities of Practice: Learning, Meaning, and Identity. CUP (1998)

Using online presence data for recommending human resources in the OP4L project

Monique Grandbastien[1], Suzana Loskovska[3], Samuel Nowakowski[1], Jelena Jovanovic[2]

[1] LORIA – Université de Lorraine - Campus Scientifique- BP 239
F-54506-Vandoeuvre les Nancy Cedex – <first name.name@loria.fr>
[2] FOS-Faculty of Organizational Sciences, University of Belgrade
Jove Ilica 154 - Belgrade, Serbia, <jeljov@gmail.com>
[3]Faculty of Computer Science and Engineering - Ss. Cyril and Methodius, Rugjer Boshkovikj,
Skopje, Macedonia <suzana.loshkovska@finki.ukim.mk>

Abstract.
Web-based Personal Learning Environments (PLEs) have been widely recognized as a mean for supporting and assisting online learning practices. A PLE is a set of services customized by the student. Among these services, resource (either digital or human) recommendation is a crucial one. The paper briefly reviews existing approaches for recommending resources in PLE. Then it describes a novel approach that relies on students' social presence data and is implemented in the OP4L prototype. OP4L makes use of ontologies to formally represent and make use of the students' social presence data. Then the paper reports on qualitative studies that were aimed at getting students' feedback about the social-presence-aware services offered by the OP4L prototype.

Keywords. Web-based learning, social presence, online presence, ontology based resource recommendation.

1   Introduction

Personal Learning Environments (PLEs) have been widely adopted in the TEL research community as a mean for facilitating learning practices. From the technical perspective, a PLE is a customizable set of services aimed at enhancing the learning experience and learning outcomes. Among these services, resource (either digital or human) recommendation services are crucial, given the number and the diversity of available resources on the Web. Various approaches to the recommendation of resources have been proposed [1-3]. They all rely on a learner profile and include a more or less rich description of the learning context, often based on ontologies.

The recent increase in the use of social software tools by learners lead to the inclusion of novel forms of social presence into PLEs. These include online status updates, online visibility, availability for online communication and the like. Semantic Web technologies, ontologies in particular, allow for taking these forms of social presence into account when generating recommendations for students.

This paper reports on qualitative studies that were conducted with students using a PLE prototype developed in the scope of the OP4L (Online Presence For Learning) project [4]. First, we briefly summarize the results from some previous studies that explored PLEs, social presence and recommendation of resources in the context of online learning. Then, we present the OP4L framework with a focus on its social presence features. Finally, we describe how we organized data collection to get an initial feedback from students and discuss the obtained results.

2 Background

In her "vision" paper [5], Vassileva defines three main roles to be performed by PLEs: *(1) support the learner in finding the right content (right for the context, particular learner, specific purpose of the learner and pedagogically), (2) support learner to connect with the right people (…) and (3) motivate/incentivize people to learn.*

To achieve these goals, researchers and developers build on experiences gained from several domains. The discovery and retrieval of learning resources is one of them and has been widely investigated, beginning with the work on metadata interoperability, then going on with the use of ontologies to better match the learners' needs and context. As social web applications, such as resources tagging, became available, solutions mixing both ontology and tagging-based approaches were proposed. Meanwhile the recommender systems community developed powerful algorithms for the e-commerce sector and PLE developers tried to adapt them to e-learning purposes [1-3].

Social presence has been identified as a crucial success factor in e-learning for many years [6]. At the beginning social presence was mostly implemented through online forums and Instant Messaging tools that allowed establishing and maintaining social presence in online learning settings. The wide adoption of social web applications resulted in the inclusion of online social networks and connections established in these networks into online learning environments. Though in theory students can interact with their entire social network, in practice they do not get any indicator about who is really available in the given moment and who is really competent for helping in the current task. Although recommending knowledgeable people for performing a given task is not new, it has been mostly investigated in company settings such as reported, for instance, in [7]. OP4L project proposes solutions for these last two challenges as described below.

3 OP4L framework

3.1 Background and objectives

OP4L is a European SEE-ERANET project which aims at exploring the use of web tools and services for supporting social presence in online learning environments and thus contributing to an improved learning experience [4]. In this paper, we use OP4L to name both the project and the developed prototype.

OP4L defines online presence as a temporary description of a user's presence in the online world. It can be considered as an image that a person projects about

him/herself into the online world. In this project we explored online presence in the context of the DEPTHS PLE [8]. DEPTHS (DEsign Patterns Teaching Help System) is designed for a Design Pattern course in Software Engineering – Computer Science – master level. It makes use of ontologies as a common base for the integration of different systems and services in a common environment for collaborative learning of software design patterns. In addition, the ontologies served as the foundation for the development of the DEPTHS' recommendation services. The first service is a context-aware recommendation of resources on software design patterns from online repositories, learning artifacts produced and shared by peers, software projects, discussion threads, chats, etc.; the second service is a context-aware recommendation of other students, experts and/or teachers to offer help in the given situation.

Within the OP4L project, DEPTHS' services have been extended to include the notion of online presence. The novel online-presence-aware educational services make use of users' online presence data when providing learners with recommendation on whom to ask for help or collaborative work. These data are periodically "pushed" towards the PLE by specific software modules developed for that purpose. Within the online presence data, a key indicator is the "online status" as declared by the user. For instance, a peer whose online status indicates that he/she is busy in the given moment will not be recommended; on the other hand, the system would recommend a face-to-face study session with a peer who has just checked in the same building and whose status indicates that he/she can be freely contacted.

A complete technical description of the OP4L models can be found in [9],

3.2   Main features of OP4L prototype

OP4L services are accessible through a dedicated Moodle platform and become available after a student selects a course to study and a learning activity. Specifically, the system indicates who is competent and available online for help or collaboration and how to contact potential helper(s)/collaborator(s), either on the Moodle platform itself, or via Facebook or Twitter. The platform also recommends appropriate content related to the topic of the course. For enhancing collaboration, students are also given a brainstorming tool where ideas can be annotated and rated. Finally, students can upload their work on the platform and benefit from a system of peer evaluation. They can assess other proposals only when they have uploaded their own solution.

4   First feedbacks from students

Two studies were run between January and May 2012 with the first versions of the OP4L prototype: The first one with Human Sciences students in Nancy (France), the second one with Computer Science students in Skopje (Republic of Macedonia).

4.1   Experiment at Université de Lorraine (Nancy, France)

Our objective was to get an early feedback about OP4L services from students to analyze how end-users use, like/dislike, and benefit from the newly provided features.

The study was conducted with 15 students in February 2012. The participants came from several master courses on different subjects. This diversity was important for the project as we needed to get feedback about recommendation services from students studying different subjects in several countries. The students' profiles were: Bachelor in Communication (8), Master in Laws and European right (3), Master in Chemistry (1), Master in Digital Design (2) and Bachelor in Medical Sciences (1).

The data collection was organized in three steps. A first questionnaire was passed with the aim of getting descriptive data about students' understanding and current use of Web-based social networking services. Then the OP4L's online presence services were demonstrated; the students were free to analyze more deeply the services. Finally, a second questionnaire was passed to get the students' feedback about the features of online presence services. To learn more about their expectations from such services, they were also asked to describe a scenario including the kind of services they would dream about. They were also invited to provide free suggestions.

Preliminary conclusion. The hypothesis was that providing students with online presence recommendation services in a LMS could significantly help them in performing their learning tasks. Students immediately show an interest in contacting peer students for the kind of project they have to complete for their master degree. Some comments: "This tool can be useful to help us identifying the appropriate additional contents, to collaborate on specific topics and to get advices on the already done work. There is a true social aspect (peer-to-peer) which could really help us." "This tool could give the possibility to ask questions and to collaborate with other students who have the capability and the availability to answer to questions. To have all contacts and friends in the platform will bring a gain of time". "To create and display detailed users' profiles including curricula, centers of interest and whom they helped and in which domains. This gives us a possibility to have a better knowledge of the people who could be contacted for help".

The analysis of the students' current use of communication technologies shows that they are not daily users of the university's Moodle environment. However, they join each other preferably through social networks. Given that context, their appreciation of the prototype ("The overall ergonomic design is quite good, clear and well organized. It is really user friendly.") is quite encouraging as they immediately understand the benefits that such a tool could bring to them in the tasks they have to achieve. Moreover we collected their appreciations (ranking) about several dimensions of online presence, knowing who is online and who is available are the most appreciated. They also mention several times a gain of time in achieving their tasks.

4.2 Evaluation at Ss. Cyril and Methodius University, Skopje (Macedonia),

The objective of this evaluation study was to provide evidence that the OP4L framework can be successfully used by students for collaborative problem solving activities. The study scenario was project-based learning with collaborative learning support where a teacher defines a specific problem to be solved in a workshop-like manner. To perform collaborative learning activities, students had to use OP4L services:

peers recommendation (suggestions of other students, teachers or experts as possible collaborators) and recommended reading (suggestions for relevant learning content to be consulted and/or used when working on the problem's solution). The evaluation study was performed at the Faculty of Computer Science and Engineering, between February and May 2012. Two groups of students participated in the study: 1) 22 master students enrolled in the Design Patterns course; these students were obliged to do two of their five course homeworks using the system; and 2) 14 undergraduate students (2nd and 3rd year) enrolled in the Human-Computer Interaction course; these students had either limited or no knowledge of software design patterns.

The study scenario was the same for both groups; the difference between the groups was in the tasks they were required to perform. At the beginning all students filled in a motivation questionnaire (MSLQ). Then, the students performed two tasks related to Design Patterns and for each task they had to do the following: propose and submit an idea for the task solution; discuss and grade ideas proposed by the colleagues; propose a solution in the form of an UML model; and assess his/her own solution and solutions of others. At the end, each student filled in a questionnaire related to the study tasks. This questionnaire was concerned with: the habits of using Facebook, the experience in using the OP4L learning environment, collaborative learning and the use of Facebook in collaborative learning.

Students were positive about the following system' services: 1) possibility to discuss with peers: 65.5% stated that discussion with peers helped them to rethink their proposal for the solution; and 2) recommending reading: 65.5% reported that reference materials helped them to rethink their proposal for the solution.

However, the results also showed that the students are not keen to use Facebook, its chat and applications for learning. Only 8 (22.2%) of them reported that they used OP4L's Facebook connection, whereas 18 (50%) students explicitly stated that they did not use the OP4L's Facebook connection. Students answers related to the use of Facebook can be grouped as follows: seven students did not have a Facebook account; those who had Facebook account, usually use the Facebook for fun or for "private matters"; some students did not use OP4L's Facebook connection because they had access to the solutions of others, so for them it was enough to solve the problem; and one student reported that the complete use of the system was too difficult and that was the reason why he/she did not try OP4L's Facebook connection.

Preliminary conclusion. Considering the answers in the questionnaire we can conclude that some of developed services for OP4L are positively assessed by students. Generally, the students as the most positive assessed the services recommending reading and "offline" discussion with peers (providing ideas for solution, and commenting and assessing the ideas and the solutions).

The use of Facebook, its chat and applications did not increase after period of working with the system. It was an unexpected result since the students tend to use the Facebook chat to communicate with the teaching staff on the course-related matters. This evaluation showed that the majority used their Facebook accounts as they had used before the study, which mainly includes fun and social communication.

5   Conclusion and further work.

We have presented online-presence-aware recommendation services as implemented in the DEPTH PLE, as well as the first assessment of these services with students. The presented studies only aimed at providing a qualitative analysis of the services and the PLE in general. Quantitative evaluations of these services are currently under way by other project partners.

Next steps include improving the software solution to deploy it in more universities and to provide teachers with ways to add new lessons and new courses. Indeed, the recommendation of resources for a given task relies on the availability of task domain descriptions through ontologies. Providing the corresponding ontologies for a new domain as well as describing resources (digital and human) is a time-consuming task. So, there is a need for an intermediate solution. The students' ratings of different dimensions of online presence and the services effectively used could help us to choose the appropriate services to implement. Then further evaluation in larger learning settings (longer period in the academic year, lower dependency to exam ratings, etc.) should take place.

6   References

1. Vuokari, R., Manouselis, N., Duval, E. eds., Special Issue on Social Information Retrieval for Technology Enhanced Learning, Journal of Digital Information, 10 (2), (2009)
2. Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., Koper, R.., Recommender Systems in Technology Enhanced Learning. Handbook of Recommender Systems. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds). 387-415, (2010)
3. Santos, O.C., Boticario, J.G.. Educational Recommender Systems and Technologies. Practices and Challenges. IGI Global, (2012).
4. OP4L project's website: `http://op4l.fon.bg.ac.rs/`
5. Vassileva, J., Towards Social Learning Environments, IEEE TLT, 1 (4), 199-214, (2008)
6. Cob, S. C., Social Presence and Online Learning: A current view from a Research Perspective, Journal of Interactive Online Learning, 8 (3), (2009)
7. Beham, G., Kump, B., Ley, T., Lindstaed, S. N., Recommending knowledgeable people in a work-integrated learning system, 1st RecSysTEL workshop, Procedia Computer Science 1, 2783-2792, Elsevier, (2010)
8. Jovanović, J., Gašević, D., Stanković, M., Jeremić, Z., Siadaty, M., "Online Presence in Adaptive Learning on the Social Semantic Web," In Proc. of the Workshop on Social Computing in Education, Vancouver, BC, Canada, pp. 891-896, (2009)
9. OP4L D3.1, OP4L Models, http://op4l.fon.bg.ac.rs/sites/default/files/OP4LD3.1.pdf

An Overview of Usage Data Formats for Recommendations in TEL

Katja Niemann, Maren Scheffel, Martin Wolpers

Fraunhofer FIT, Schloss Birlinghoven, 53754 Sankt Augustin, Germany

{katja.niemann, maren.scheffel,
martin.wolpers}@fit.fraunhofer.de

Abstract. Recently, a number of usage data representations have emerged that enable the representation of user activities across system and application boundaries. Based on these user activity data, systems can adapt to the users and provide personalized information. A lot of usage data representation formats are already successfully used in real world applications. However, dependent on the purpose, the formats show different advantages and disadvantages one must consider when choosing a format for a system. In this paper, we will present the four most commonly used data representations, namely Contextualized Attention Metadata, Activity Streams, Learning Registry Paradata and NSDL to alleviate the selection of a suitable format.

Keywords: usage data formats, technology enhanced learning

1   Introduction

Attention or Usage Metadata represent the activities of users and their usage of data objects in specific applications. Aggregating and analysing the usage data provides the basis for advanced user support systems, e.g. learning recommendation or self-reflection support. Furthermore, usage data can be employed for annotating data objects with information about their users and usages, thereby rendering possible object classifications according to use frequency, use contexts and user groups [1], [2] [3].

Particularly in the domain of learning analytics (see [4], [5] and [6] for more information on learning analytics) and educational data mining, usage data provide the basis for learning support systems. For example, based on an analysis of usage data, irregularities of learning behaviour of students can be identified [7] and the results of corrective activities by the teacher can be monitored. Another example of the successful application of analysing usage data in learning settings is the reflection and comparison of learning activities among students of a learning group. Here, by playing back their learning activities, students compare themselves with their fellow students and identify how to improve their learning activities. A further example of the successful use of usage data are personalized recommender systems, e.g. in the domain of learning (see [8] for more details on recent learning recommendation systems).

Recently, a number of data representation formats for usage data have emerged. In contrast to simple logging files, these representations focus on the activities of users and not on those of a system. In this paper, we will present the most prominent examples, namely Contextualized Attention Metadata, Activity Streams, Learning Registry Paradata and NSDL Paradata.

2 Usage Data Formats

2.1 Contextualized Attention Metadata

The CAM scheme [9] was defined as an extension of Attention.XML [10] which is an early approach to capturing and storing attention metadata for single users. In the current CAM version[1], the focus has moved from the user and the data object to the event itself. This is due to the insight that not every event has a fixed set of attributes.
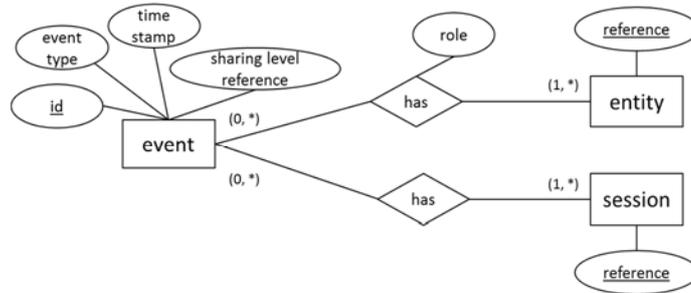

Fig. 1. CAM scheme

Additionally, only the basic information about an event is stored, e.g. the event type and the time stamp. All other information, e.g. metadata describing users or documents involved in the event, are linked. In this way, each entity and also each session can be described in a different and suitable way and no information is duplicated.

Fig. 1 shows the complete CAM scheme. The main element of a CAM instance is the *event* entry which comprises its *id*, the *event type*, the *timestamp*, and a *sharing level reference*. Examples for *event types* are "send", "update" or "select". The *sharing level reference* points to a description of the specific sharing level which describes the privacy related issues of the event. Depending on the event, various entities with different roles can be involved, e.g. when sending an e-mail, there is a person with the role sender, at least one person with the role receiver and a document with the role e-mail. Each event can be conducted in a session. A *session* can, for example, be the time between booting and shutting down a computer or the time between the login and logout of a user in a portal.

The current CAM scheme does not have fixed bindings so far. The information can be stored in XML, RDF, JSON or in a relational database, depending on the purpose of the data collection.

---

[1] https://sites.google.com/site/camschema/

2.2   Activity Streams

The Activity Streams specification [11] defines a format for single activities carried out by users. An Activity Stream is a collection of one or more individual activities. Usually, activities are serialized using JSON.
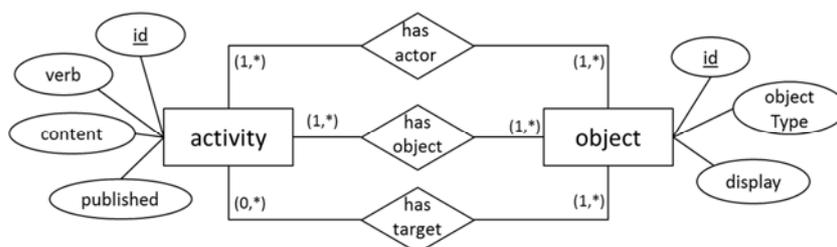

**Fig. 2.** Simplified excerpt of the Activity Streams scheme

Fig. 2 shows the core elements of the Activity Streams scheme. A single activity must at least contain a description of the entity that performed the activity (*actor property*) and the date and time at which the activity was published (*published property*). The Activity Stream Working Group recommends that an activity also contains a *verb*, an *object*, and an *id property*. The *verb* identifies the *action* that the *activity* describes (e.g. "accept", "add", "dislike" etc.), the *object property* describes the primary *object* of the *activity* (e.g. the watched movie or the sent e-mail) and the *id property* provides a unique identifier for the activity in the form of an absolute IRI (Internationalized Resource Identifier). The *target property* is optional and can be used if indicated by the verb. For instance, in the activity, "John sent an e-mail to Bill", "Bill" is the target of the activity.

The value of the *actor, object, and target property* respectively is an Activity Stream Object. An Activity Stream Object comprises several properties describing the object and should at least contain an IRI (*id property*) and a plain-text name for the object (*display property*). Additionally, it can contain others such as an *object type*. The Activity Base Schema [12] already defines *object types* to be used with Activity Streams, e.g. "alert", "application", "article", etc. The *object types* are further grouped in six classes, i.e. audio and video objects, binary objects, events, issues, places, tasks. Depending on the class, *objects* may contain further properties, e.g. *startTime* and *attendedBy* for Events. Furthermore, any object within an Activity Streams object can be extended with properties not defined by the core Activity Streams' specification to provide as much flexibility as possible.

2.3   Learning Registry Paradata

The Learning Registry Paradata format [13] is basically an extended and altered version of the Activity Streams JSON format. It was defined to store aggregated usage information about resources. The Learning Registry Paradata specification states explicitly that the Activity Streams format should be used if mainly individual actions are stored.

As for the Activity Streams, a basic LR Paradata statement consists of three key elements: *actor*, *verb*, and *object* (see Fig.3). The *actor* refers to the person or group that does something and is represented by a string or LR Paradata object (as defined later). The *verb* refers to the action that is taken. In its simplest form, it just contains the *action* name (e.g. "taught" or "viewed"), but it can also be specified in more detail, which is the main difference of the LR Paradata and the AS scheme. The *object* refers to the thing being acted upon using a string or a LR Paradata object.
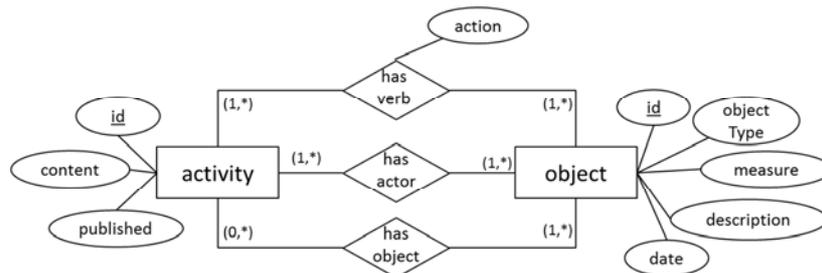
Fig. 3. Simplified excerpt of the Learning Registry Paradata scheme

A LR Paradata object may contain an *id*, an *objectType*, a *description* of the object, i.e. an array of *keywords*, a *measure* related to the object, a *date* and a *context*. Apart from *description* and *date*, each element can be represented by a string or by a JSON object without pre-defined scheme. The values of the elements depend on the *objectType*, which can be e.g. a person, a group, a learning resource, a LMS, etc. Within the *verb*, an action is specified that holds the verb's value (*action*), additionally, it can contain any element specified for the LR Paradata object [14], [15].

2.4 NSDL Paradata

The NSDL Paradata format was defined to capture aggregated usage data about a resource (e.g. "downloaded", "favourited", "rated") which is designated by audience, subject or education level [16]. In contrast to the other usage data formats presented so far, this format is not event, but object-centric. Each data object has exactly one NSDL Paradata record, which is identified by a *recordId* and must contain the URL of the resource to which the paradata record applies (*usageDataResourceURL*). The most important element is the *usageDataSummary*, which comprises all available usage statistics/information about a resource using five different types of values. An *Integer/Float* value represents the number of times certain actions have been performed on the resource, e.g. how often it was viewed or downloaded. A *String* value is a textual value that has been associated to the resource, e.g. a comment. A *RatingType* value is the numerical average that represents the judging of a resource on a numerical scale, e.g. a rating according to its usability. A *VoteType* value represents the number of positive and negative responses to a resource, e.g. good or bad for use in classroom. A *RankType* value represents the standing of a resource in a hierarchy, e.g. best of 2010.

Besides its type and value, each *usageDataSummary* element contains the beginning and ending date for the usage data (*dateTimeStart*, *endTimeStart*), information about the *audience* that conducted the event ("educator", "student", "general public", the *subject* of the used resource (e.g. "computing" or "mathematics") and in which educational level (*edLevel*) the resource was used (e.g. "MiddleSchool", "Grade 7").
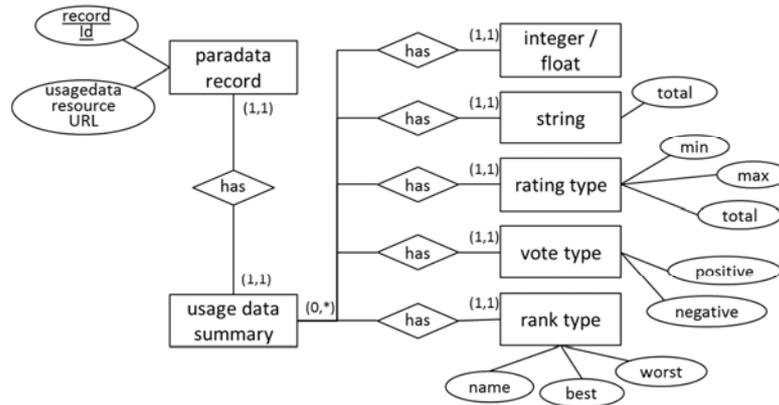

Fig. 4. Simplified excerpt of the NSDL Paradata scheme

For lack of space, these elements are not shown in Fig. 4, but only the elements that are dependent on the type of the *usageDataSummary* element. Please see http://ns.nsdl.org/ncs/comm_para/1.00/records/planets.xml for an extensive example.

3   Conclusion

We reviewed the four most popular usage data representation formats that are being used in the learning domain in this paper and described their main properties. Each format has been created with a specific purpose in mind, so one must be clear about the further applications that will use the collected usage data when choosing the most suitable format.

In order to enhance the interoperability among usage data analysis tools and usage data storage silos, our next step will be to provide guidelines on how mappings between formats can be implemented and what has to be considered. All formats are open and allow supplemental, not pre-defined elements. Additionally, the specified vocabularies are not perceived as complete and for instance in a CAM instance, an *entity* can be described by any metadata scheme. Thus, no one-size-fits-all mapping among the formats is possible. In contrast, mapping can only be defined for specific application scenarios. By providing automatic mapping rules based on specific application scenarios, access to usage data collections will be facilitated. Nevertheless, further work remains to be done in terms of further generalizing the mapping rules so that the automatic conversion tools become less application scenario dependent.

References

1. Schuff, D., Turetken, O., D'Arcy, J., and Croson, D. 2007. 'Managing E-Mail Overload: Solutions and Future Challenges', *Computer*, 40 (2), 31-36.
2. Hauser, J. R., Urban, G. L. Liberali, G., and Braun, M. 2008 'Website Morphing', *Marketing Science*.
3. Adomavicius, G., Tuzhilin, A. 2005. 'Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions', *IEEE Transactions on Knowledge and Data Engineering*, 17 (6), 734-749.
4. Elias, T. 2011. 'Learning Analytics - Definitions, Processes and Potential', retrieved July 4, 2012 from http://learninganalytics.net/LearningAnalyticsDefinitionsProcessesPotential.pdf
5. Dawson, S. 2011. 'Analytics to Literacies: Emergent Learning Analytics to evaluate new literacies'. *Workshop on New Media, New Literacies, and New Forms of Learning*, London, December 2011. Retrieved July 4, 2012 from http://blogs.ubc.ca/newliteracies/files/2011/12/Dawson.pdf
6. Ferguson, R. 2012. 'The State of Learning Analytics in 2012: A Review and Future Challenges'. *Technical Report KMI-12-01*, Knowledge Media Institute, The Open University, UK. Retrieved July 4, 2012 from http://kmi.open.ac.uk/publications/techreport/kmi-12-01
7. Scheffel, M., Niemann, K., Leony, D., Pardo, A., Schmitz, H.-C., Wolpers, M., Delgado Kloos, C. 2012. "Key Action Extraction for Learning Analytics". Proceedings 7th European Conference on Technology Enhanced Learning (EC-TEL 2012), Springer.
8. Katrien Verbert, Nikos Manouselis, Xavier Ochoa, Martin Wolpers, Hendrik Drachsler, Ivana Bosnic, Erik Duval. 2012. "Context-Aware Recommender Systems for Learning: A Survey and Future Challenges," IEEE Transactions on Learning Technologies, vol. 99, no. PrePrints, , 2012
9. Schmitz, H.-C., Wolpers, M., Kirschenmann, U., Niemann, K. 2012. 'Contextualized Attention Metadata'. *Human Attention in Digital Environments*, Eds: Claudia Roda, Cambridge University Press, Cambridge, US, 2012 http://www.cup.es/catalogue/catalogue.asp?isbn=9780521765657
10. Çelik, T. 2005. 'Attention.xml Technology Overview', retrieved July 4, 2012 from http://tantek.com/presentations/2005/01/attentionxml.html.
11. Snell, J., Atkins, M., Norris, W., Messina, C., Wilkinson, M., Dolin, R. 2012. JSON Activity Streams 1.0, retrieved July 4, 2012 from http://activitystrea.ms/specs/json/1.0/
12. Snell, J., Atkins, M., Recordon, D., Messina, C., Keller, M., Steinberg, A., Dolin, R. 2012. Activity Base Schema (Draft), retrieved July 4, 2012 from http://activitystrea.ms/specs/json/schema/activity-schema.html
13. Paradata Specification V1.0. Retrieved July 4, 2012 from https://docs.google.com/document/d/1IrOYXd3S0FUwNozaEG5tM7Ki4_AZPrBn-pbyVUz-Bh0/edit?hl=en_US&pli=1#
14. Paradata Cookbook, V 1.0 © Copyright 2011, Learning Registry. CC-BY-3.0. Retrieved July 4, 2012 from https://docs.google.com/document/d/1lggCnowWsDgQxrNjYRAgh2KNwKfq-MV8vLJzRXbAaos/edit?pli=1#
15. Paradata in 20 Minutes or Less, V 1.1 © Copyright 2011, US Department of Education: CC-BY-3.0. Retrieved July 4, 2012 from https://docs.google.com/document/d/1QG0lAmJ0ztHJq5DbiTGQj9DnQ8hP0Co0x0fB1QmoBco/edit?hl=en_US&pli=1#
16. NSDL Paradata, 2012. Retrieved July 4, 2012 from https://wiki.ucar.edu/display/nsdldocs/comm_para+%28paradata+-+usage+data%29