Wong, L.-H. et al. (Eds.) (2013). Proceedings of the 21st International Conference on Computers in Education. Indonesia: Asia-Pacific Society for Computers in Education

Competence Analyser: A portable GUI tool for modelling domain and learner knowledge

Simone Kopeinik[a*], Michael Bedek[a], Georg Öttl[a] & Dietrich Albert[ab]
[a] *Knowledge Technologies Institute, Graz University of Technology, Austria*
[b] *Department of Psychology, University of Graz, Austria*
*simone.kopeinik@tugraz.at

Abstract: Learner models form the basis of adaptive learning systems, representing what the system knows about a learner. Knowledge about a learner's expertise within a learning domain plays an essential role when recommending learning objects as well as when supporting a learner's reflection and awareness. In this paper we present the Competence Analyser tool to construct knowledge structures as a foundation for learner models in accordance with a psychologically sound framework, the Competence based Knowledge Space Theory (CbKST), which supports performance assessment in user-adaptive learning systems. The tool enables the learner to actively influence the learning profile creation by setting learning goals and defining prior knowledge in the context of learning domains. A first prototype has been developed and evaluated within the frame of the EC-funded TARGET project that provides a game-based learning environment with adaptive story recommendation. The paper outlines how to foster awareness and self-regulation while building learner and domain models for adaptive learning systems using a graph-based GUI tool.

Keywords: Learner Modelling, Knowledge Modelling, Graph-based Visualization, Prerequisite-relations, Competence based Knowledge Space Theory, Critical Reflection

1. Introduction

In user-adaptive learning systems information about the learner is used to identify and recommend learning objects, or more concretely to select and adapt instructional strategies to the learner (Kay & Kummerfield, 2012). A learner's specific needs may depend on various factors, including prior knowledge (e.g. expertise levels), motivation, learning styles, learning goals and learning efficiency (Klašnja-Milićević, Vesin, Ivanović & Budimac, 2011). Some of these factors, such as knowledge or motivation, typically vary within a single learning cycle. Hence, to support adaptivity during a learning cycle the learner model needs to be continuously updated. To this end, information about the learner may be collected through a combination of implicit performance assessment and explicitly requested user input (Brusilovsky & Millán, 2007), and then strategically integrated into the learner model.

The psychologically sound framework behind our approach is the Competence-based Knowledge Space Theory (CbKST) (Heller, Steiner, Hockemeyer & Albert, 2006). The CbKST is a framework for representing the conceptual organisation of a given body of competences. It extends the behaviourist Knowledge Space Theory (Doignon & Falmagne, 2011), which focuses on the set of problems or tasks a person is able to master. Beyond that, the CbKST concentrates on the identification of latent skills and competences and prerequisite relations between them: If a person wants to develop competence x, he or she has to develop y beforehand. These competences and their prerequisite relations represent a competence structure which forms the basis of learners' domain models and, furthermore, for updates on these models.

Technology-enhanced learning (TEL) applications building upon the CbKST require the definition of the competence structure and the competences a learner exhibits and aims to develop at the very beginning of the learning process. Therefore, we introduce the Competence Analyser (CA), a tool that allows defining initial competence information needed to provide CbKST based assessment and recommendation strategies. The first part of the CA constitutes the Competence Structure Editor (CSE). The CSE embodies the interface to model the competence structure of a learning domain, consisting of competences and their interrelations (i.e. prerequisite relations). It also supports the learner in gaining insights into the learning domain in terms of critical reflection on learning goals and competences and on how these competences of the learning domain relate to each other.

The second part of the GUI is to define a learner's prior knowledge, in form of a "User Competence Profile" (UCP) and learning goals by means of a "Target Competence Profile" (TCP). The difference between these profiles results in the "Gap Competence Profile" (GCP) that encompasses the set of competences a learner aims to develop by engaging with the TEL application.

Relevant and related work to the CA can be found in European research projects in the field of TEL. APOSDLE (http://www.aposdle.tugraz.at), a work-integrated learning system, implemented two competence related authoring tools. First, Moki (Ghidini et al., 2009) allows modelling tasks and concepts (or skills) in a wiki and to export this model as an OWL-Ontology. The second competence related authoring tool developed within APOSDLE, the Task Competence Tool (TACT), allows to directly model tasks, topics and relations between tasks, topics and competences needed for a certain task (Lindstaedt & Christl, 2011). Another description about competence gap analysis was given by Coi et al. (2007) who presented a gap analysis tool on a conceptual level. Nkambou, Gauthier & Frasson (1996) amongst others, describe similar authoring system, but do not work on the basis of competences and competence relations.

In this paper we will introduce the Competence Analyser tool as a GUI component to define CbKST supporting user and domain model construction. We will present the graphical user interface including information about its implementation and possible application. Subsequently, we will provide insights into the main results of the usability study and finish with a brief discussion on the topic.

2. Functionality and implementation

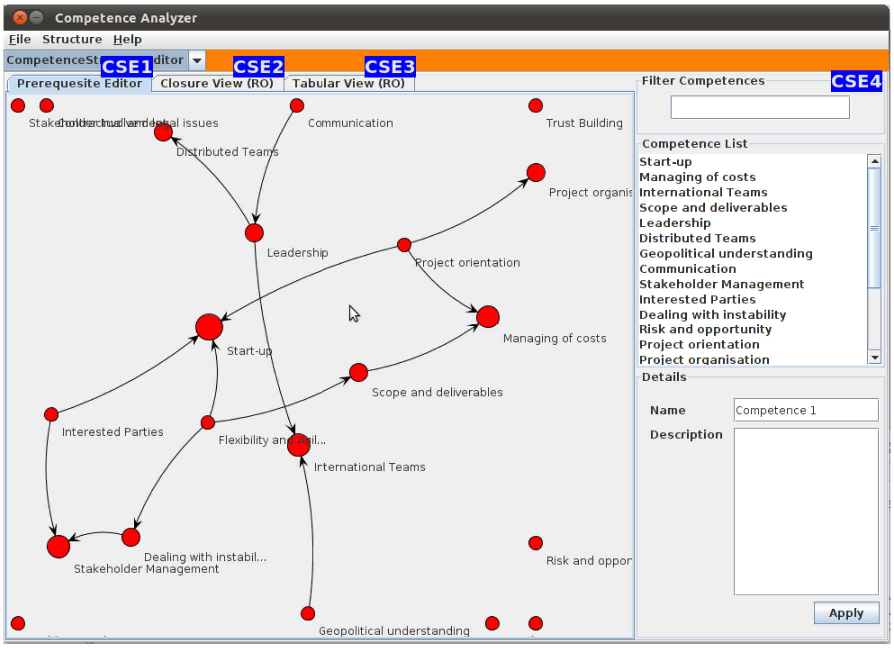
A first prototype of the CA was implemented for and integrated in a Digital Educational Game in the course of a TEL platform called TARGET (www.reachyourtarget.org). The main component of TARGET is a 3D virtual learning environment which consists of story dependent indoor and outdoor locations where the learner communicates with non-playable characters to gather pieces of information in order to master game scenarios. Stories represent learning objects that are recommended to a user on the basis of a learner's current competence state modelled and updated through the CbKST service that works on the basis of the models designed in the CA (Kopeinik, Nussbaumer, Bedek & Albert, 2012).

In TARGET each learning cycle was described through a learning plan (i.e. an ordered set of game scenarios the learner has to engage with). We developed CA to assist in the definition of learning plans with learning goals, prior knowledge and a learning domain. Within the two main frames of the GUI learner and instructor can handle tasks such as: (1) the definition of the learning domain in terms of competences and relations between competences (Competence Structure Editor) and (2) the selection of competences to learn.

2.1 Competence Structure Editor

The Competence Structure Editor (CSE) was implemented as a prototype to support instructors and organizations in creating competence structures. In this context, a competence structure is defined as a set of competences ordered by means of prerequisite relations. A prerequisite relation between two competences x and y means: "If a person wants to develop competence x, he or she has to develop y beforehand" which is conceptually the same as "If a person has shown that he or she has competence x, he or she has competence y too". In this case, competence y is a prerequisite competence for competence x.

The CSE provides not only a tool to relate competences to each other but also to visualize a previously constructed competence structure as a modified Hasse diagram. The main difference to a standard Hasse diagram is the visualization of the connectedness of a vertex by increasing the diameter of the vertex (see Figure 1). This visualization should make the information contained in the competence structure more meaningful and insights more intuitive. The Hasse diagram is made directly editable by mouse events. A mouse editing event is only accepted if it does not contradict the Hasse diagram. A contradiction can e.g. occur if a cycle is introduced or by trying to add edges which are already indirectly inferred by the Hasse diagram. For the cycle detection we used an algorithm which is based on a dynamic topological sort algorithm as described by Pearce and Kelly (2006). For calculating the transitive reduction of the graph as needed by the Hasse diagram visualization, we implemented a simple algorithm which basically calculates the transitive closure of the graph (Aho, Garey & Ullman, 1972). Consequently, for every valid editing action the transitive reduction and the topological order of the underlying graph data structure is updated. The visualization itself was based on the Jung Framework (Madadhain et al., 2005).



Name: Competence 1

Figure 1. The Competence Structure Editor GUI.

The CSE (see Figure 1) offers three different views: (CSE1) The Prerequisite Editor which allows to create and edit competences and to model their interrelations. (CSE2) The Closure View displays the competence structure as a more detailed representation that also includes transitive connections. (CSE3) The Tabular View displaying the competences and their relations in form of a table. (CSE4) Filter and Edit Panels are part of all three views and can be used to search and edit competences.

2.2 *Competence Analyser*

The CA view fosters a learner’s active part in the personalization process. It is provided as an input device to create an initial state of the learner profile and is used either by the learners and their supervisors in collaboration, or by the learners themselves. Using the CA the learner is asked to drag and drop competences from the list provided on the left hand side to either the UCP pane (information about existing expertise) or the TCP pane (learning goals). When the learner is finished, the UCP pane should include all competences the learner is competent in the use of at this point in time. The TCP should include all competences the learner would like to show after the execution of the resulting learning cycle. The Gap Competence Profile (GCP) then shows the competences a learner has to learn in order to achieve the defined goal state.

Figure 2 presents the visual framework of the CA: (CA3) The Competence Panel implements a tree like representation of the competence structure that is based on the inferences available from the Competence Structure Editor and the semantics of a Prerequisite Relation. The competence panel provides drag and drop functionality from the panel to a visualization. (CA4) The Profiles Pane shows predefined job profiles consisting of a set of competences. (CA5) The User Competences Profile Pane allows adding and editing of competences that are identified as prior knowledge of a learner. (CA6) The Target Competence Profile Pane allows adding and editing of competences that are defined as the learning goal. (CA7) A panel that visualizes the difference between TCP and UCP. (CA1, CA2, CA8 and CA9) Menu Items to save, load and update data in the Panes.

Some options offered by the interface are: (a) Drag and drop of competences from the list view to the profile tabs. (b) Selection and multiple selection of competences to move/delete within the profile tabs; (c) The right mouse click on a competence-vertex displays a context menu; (d) Scrolling is supported using the mouse wheel.

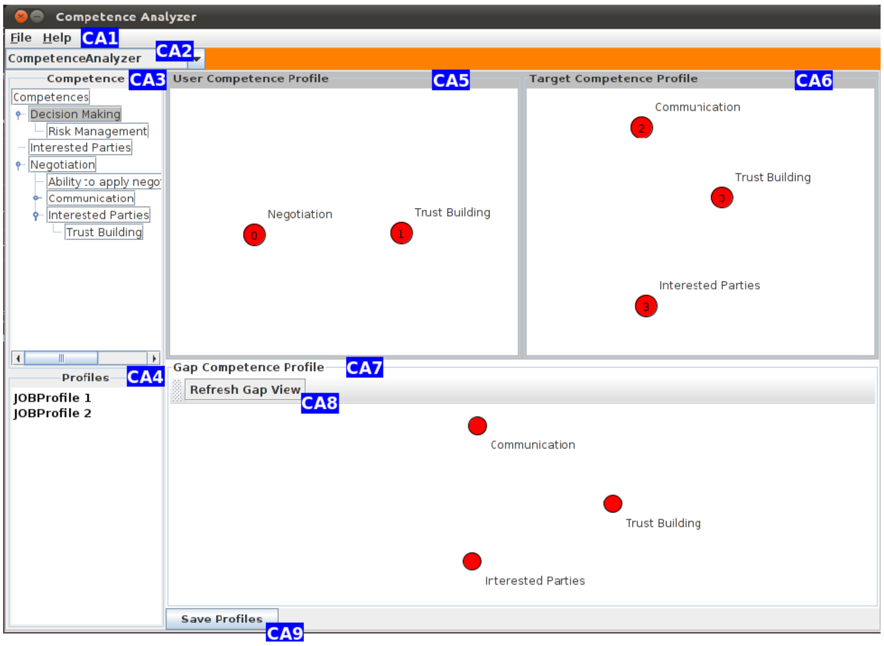


Figure 2. The Competence Analyser GUI.

3. Usability Studies

During the development process of the Competence Analyser we carried out several iterative evaluation cycles for both, formative and summative evaluations, respectively. For the formative evaluations we applied a thinking-aloud method with a small set of experts (n = 4 – 6) in the field of cognitive psychology and TEL. Participants were asked to verbalize their thoughts as they interacted with the tool. For each major release of the tool, we applied a more standardized procedure for carrying out summative evaluations. The rationale behind a more standardized procedure was to compare the evaluation results over time (at least on a descriptive level). For each of the studies, a minimum of eight participants who were unfamiliar with the tool took part. The participants have been provided with the background and purpose of the evaluation study as well as with a brief introduction to the tool´s interface and its main features. Afterwards they had to structure a learning domain (in the field of elementary algebra) and define a current and target competence profile for an imaginary 2nd grade pupil. Finally, they were asked to fill out a short version of IsoMetrics inventory for the testing the usability of the GUI (Willumeit, Gediga & Hamborg, 1996). IsoMetrics is based on the ISO 9241/10 standard. The short version of the IsoMetrics inventory consists of the following 7 factors:

- Suitability (15 items), e.g.
  “In a given screen, I find all of the information I need in that situation.”
- Self Descriptiveness (12 items), e.g.
  “The terms and concepts used in the software are clear and unambiguous.”
- Controllability (11 items), e.g.
  “In order to perform my tasks, the software requires me to perform a fixed sequence of steps.” (formulated conversely)

- Conformity with User Expectations (8 items), e.g.
  "When executing functions, I have the feeling that the results are predictable."
- Error Tolerance (15 items), e.g.
  "I have never made an entry that caused a software error (e.g. a system/program crash or an undefined dialog state)."
- Suitability for Individualization (6 items), e.g.
  "The software lets me adapt forms, screens and menus to suit my individual preferences."
- Suitability for Learning (8 items), e.g.
  "In order to use the software properly, I must remember a great many details."
  (formulated conversely)

The rating scale for the IsoMetrics inventory has a range from 1 (predominantly disagree) to 5 (predominantly agree) as well as the opportunity to rate 0 (no opinion). The following figure shows the results (means and standard deviations for all sub-scales) for the final summative evaluation study with a sample size of 9 participants.

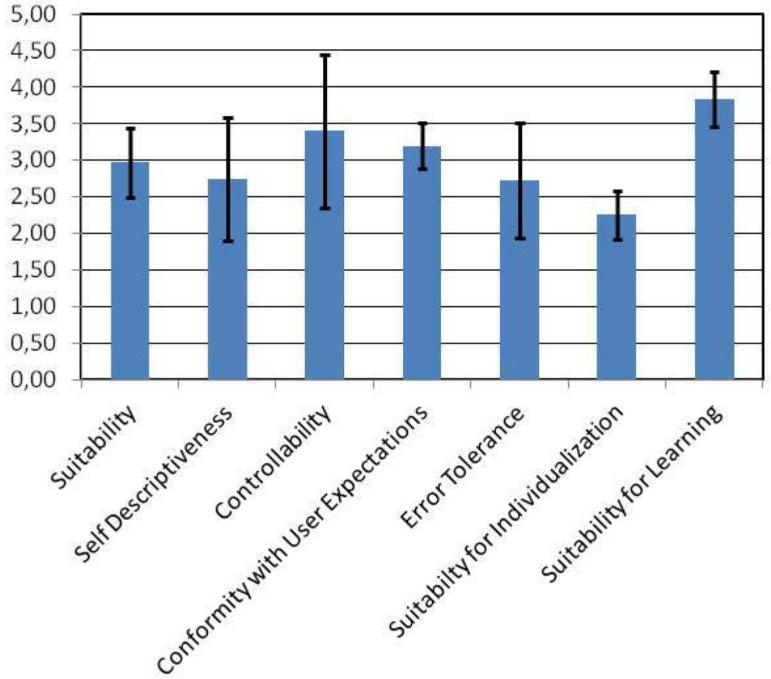


Figure 3. IsoMetrics – results of the summative evaluation study

As shown in the figure above, the results of the formative evaluation indicate, that the Competence Analyser Tool has potential for improvement. However, the quite high value for the dimension *suitability for learning* is very satisfying. In any case, we consider the results in general and the answers / responses to the open questions of the short version of the IsoMetrics inventory, as important insights that will be taken into account during further iteration cycles in the development of the Competence Analyser Tool.

4. Discussion

The Competence Analyser was developed, integrated and repeatedly evaluated in the course of the European research project TARGET (7th framework program), where resulting learning profiles served as a basis for adaptive learning object recommendation (Kopeinik, Nussbaumer, Bedek & Albert, 2012). Additionally, it aims to facilitate the learner´s reflection upon his or her own performance, strengths and weaknesses: The CA-GUI is the application of an open learner model (Bull, 2004), where the learner´s interactions with the GUI and his or her reflection should lead to an enhanced learning process (Mitrovic & Martin, 2002). We will extend the reflection support of the tool by editing the appearance of nodes according to the learning performance of completed learning tasks.

Although the tool worked well from a technical perspective, usability studies have shown that some end-users found it difficult to handle the bubble visualization in the user profile tab and would have preferred a simpler table like view. Unfortunately, the summative evaluation studies were restricted to one time use (e.g. different participants were in the samples) and therefore do not consider potential improvements after familiarization. Problems have also been identified with the visualization of the knowledge domain, since readability tends to decrease with the growth of the graph. The main issue is the increasing number of edge crossings which hinder visual clarity. Future work will therefore inspect algorithms to optimize the positioning of nodes in order to minimize edge crossings.

Acknowledgements

This paper is part of the EC-Project TARGET funded by the 7th Framework Programme of the European Commission’s IST-Programme (contract no. 231717). The authors are solely responsible for the content of this paper. It does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of data appearing therein.

References

Aho, A.V., Garey, M.R., & Ullman, J.D. (1972). The Transitive Reduction of a Directed Graph, *SIAM Journal on Computing*, *1*, 131-137.

Brusilovsky, P., & Millán, E. (2007). User models for adaptive hypermedia and adaptive educational systems. In P. Brusilovsky, A. Kobsa and W. Nejdl (Eds), *The Adaptive Web* (pp. 3-53). Berlin, Heidelberg: Springer-Verlag..

Bull, S. (2004). Supporting learning with open learner models. In: Proceedings of the 4th Hellenic Conference on Information and Communication Technologies in Education, Athens, Greece, September 29 - October 3, pp. 47-61.

Coi, J.L., Herder, E., Koesling, A., Lofi, C., Olmedilla, D., Papapetrou, O., & Siberski, W. (2007) A Model For Competence Gap Analysis, In *Proceedings of 3rd International Conference on Web Information Systems and Technologies* (pp. 4-6), INSTICC Press.

Doignon, J.-P., & Falmagne, J.-C. (2011). *Learning Spaces*. Berlin, Heidelberg: Springer-Verlag.

Ghidini, C., Kump, B., Lindstaedt, S., Mahbub, N., Pammer, V., Rospocher, M. & Serafini, L., "MoKi: The Enterprise Modelling Wiki, In Demo Proceedings of the 6th European Semantic Web Conference (ESWC 2009), 831-835.

Lindstaedt, S. & Christl, C (2011).APOSDLE – learn@work: Firsthand Experiences and Lessons Learned. Work-Integrated Learning in Engineering, Built Environment and Technology: Diversity of Practice in Practice. (2011), S. 52 - 71

Nkambou, R., Gauthier, G., & Frasson, C. (1996) CREAM-Tools : An Authoring Environment for Curriculum and Course Building in an intelligent Tutoring System, 3rd International Conference on Computer Aided Learning and Instruction in Science Engineering , Lecture Notes in Computer Science, no. 1108, pp. 186-194, Springer Verlag, Donostia, San Sebastian, Spain.

Heller, J., Steiner, C., Hockemeyer, C., & Albert, D. (2006). Competence-Based Knowledge Structures for Personalised Learning. International Journal on E-Learning, 5(1), 75-88.

Kay, J., & Kummerfield, B. (2012). Lifelong learner modelling. In P.J. Durlach and A.M. Lesgold (Eds), *Adaptive Technologies for Training and Education* (pp. 140-165). New York: Cambridge University Press.

Klašnja-Milićević, A., Vesin, B., Ivanović, M., & Budimac, Z. (2011). E-Learning personalization based on hybrid recommendation strategy and learning style identification, *Computers & Education*, *56* (3), 885-899.

Kopeinik, S., Nussbaumer, A., Bedek, M., & Albert, D. (2012). Using CbKST for Learning Path Recommendation in Game-based Learning. In *Proceedings of the 20th International Conference on Computers in Education ICCE 2012*(pp. 86 – 91).

Madadhain, J., Fisher, D., Smyth, P., White, S., & Boey, Y.B. (2005). Analysis and visualization of network data using JUNG, *Journal of Statistical Software*, *10*, 1-35.

Mitrovic, A. & Martin, B. (2002). Evaluating the Effects of Open Student Models on Learning, In. P. De Bra, P. Brusilovsky and R. Conejo (Eds.). Adaptive Hypermedia and Adaptive Web-Based Systems (pp. 296-305), Berlin: Springer-Verlag.

Pearce, D. J., & Kelly, P.H. (2006). A Dynamic Topological Sort Algorithm for Directed Acyclic Graphs, *Journal on Experimental Algorithmics*, *11*, 1-24.

Willumeit, H., Gediga, G., & Hamborg, K. (1996). IsoMetricsL: Ein Verfahren zur formativen Evaluation von Software nach ISO 9241/10, *Ergonomie & Informatik*, *27*, 5-12