Ontology-based User Modelling for the Semantic Web

Judy Kay, Andrew Lum

[1] School of Information Technologies
University of Sydney, NSW 2006
Australia
{judy, alum}@it.usyd.edu.au

Abstract. This paper discusses the ways in which ontologies can be useful for user modelling. In particular we are interested in the use of ontology-based user models in semantic web applications that can easily integrate evidence based reasoning from typical web data. We examine the roles that ontologies play in user modelling as well as the requirements that user modelling imposes on ontologies. We reflect on our experiences in ontology-based user modelling in an educational context.

1 Introduction

Adaptive systems often require some form of representation for their domain in order to provide a backbone for user modelling and reasoning. An ontology has the potential to fill this role. Ontology-based user modelling is the use of ontologies to structure user models. In this paper we are particularly interested in applying ontologies to existing web based systems in order to add a level a semantics that will support adaptivity and efficient user modelling.

An ontology is defined by [1] as "an explicit specification of a conceptualisation". A conceptualisation consists of a set of entities (such as objects and concepts) that may be used to express knowledge and relationships. People have their own internal conceptualisations of different domains, and can automatically adapt different conceptualisations for different contexts.

An ontology is a way to express these objects, concepts, relationships in a conceptualisation in an explicit way, for example, through logical axioms or even through a visual representation such as a graph. It forms a shared and common understanding that is intended to eliminate terminological and conceptual ambiguity. The ontology provides a basis for communication between people who have different contextual viewpoints as well as inter-operability among dissimilar systems by forming a common vocabulary, thus provides invaluable system engineering benefits [2].

Although ontologies have their roots in the field of artificial intelligence, they have also attracted the attention from other fields such as knowledge management, information retrieval and electronic commerce [3]. One reason for this is because of the shift from computers being isolated machines to being part of a larger network, there is a need for a mechanism to facilitate communication between application systems and people [2]. In particular, the Semantic Web vision [4] has been promoting the use of ontologies to provide a common language for automated reasoning about content for the World Wide Web.

There are many potential roles that ontology can play to support user modelling. Some of these are identical to the broader uses of ontologies, such as supporting reasoning across granularities, providing a common understanding of the domain to facilitate reuse, and harmonization of different terminologies. There are also some requirements specific to user modelling such as scrutability and the ability to support a reasoning layer specific to user evidence.

Section 2 of this paper discusses the roles for ontologies in user modelling and provides examples of such systems. Section 3 examines the requirements, limitations and trade-offs of ontologies and methodologies. Section 4 describes our own experiences with ontology-based user modelling followed by a discussion in Section 5.

2 Roles for Ontologies in User Modelling

We have identified three important roles that ontologies play in user modelling. These are illustrated in Figure 1. Ontologies have a role in defining user models, providing a vocabulary for metadata in the objects used in a domain, as well as in user interfaces to the user model. We will briefly describe each of these.

The user model represents beliefs about a user including, for example, their preferences, knowledge and attributes for a particular domain. One of the essential tasks in establishing a suitable user model for a domain is to establish its vocabulary. Ontologies has a clear and important role for this task. If there were an established ontology for the domain of interest, it would be natural to adopt this as a starting point for the user model vocabulary. For example, in the teaching of elementary arithmetic, the ontology would include concepts which as described in English with words like *subtrahend* and *minuend*. An ontology of this domain may include these concepts and then, if this ontology were used by all systems that deal with teaching in this domain, there could be useful sharing of knowledge about the user across systems. In addition, if we wanted to create a new teaching system in this area, we would have an excellent start to defining a suitable user model if we at least consider this as a starting point for the choice of the user model vocabulary.

Of course, another important role for ontologies is in supporting inference on the user model. For example, if the available evidence about the user is all at the fine-grained level, the ontology can usefully provide a basis for inferring about the user attributes that are at a coarse granularity. Similarly, ontologies may support combination of evidence about the user where the evidence sources use different terms and the ontology can harmonise these.

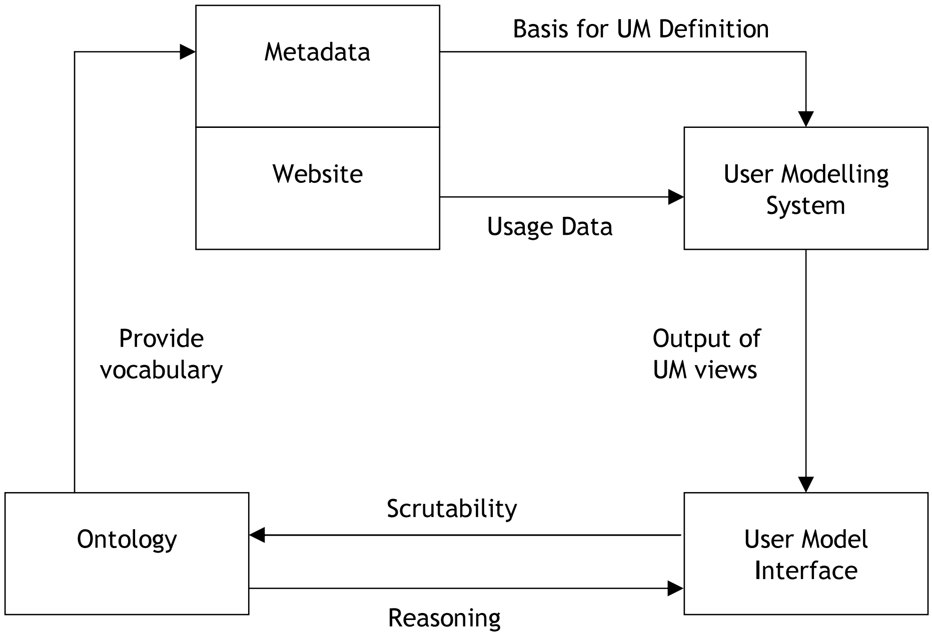

Fig. 1. Architectural diagram showing the roles of the ontology and it's interaction with other core components in adaptive web based systems

Figure 1 also shows that ontologies can play a role in the definition of metadata. Part of this parallels the discussion above because it is critical that metadata describing objects in a personalised system should have vocabulary that maps to the user model. So, for example, if a digital object teaches about *GOMS analysis*, it is important that a system be able to match this with parts of the user model to determine if the user wants to learn about *GOMS analysis*. Ontologies may also have an important role in interfaces that assist on the markup of metadata.

The figure shows a third role for ontologies in supporting the interface to a user model. Essentially, this role follows from the fact that most visualisations are based upon a graph structure. In cases where the user model does not have a a suitable graph to organise the components in the nodel, an ontology should be extremely useful, both because it can provide such a graph structure and because that structure should make sense to the user, in terms of the meanings of the concepts modelled.

3 Desirable Properties and Trade-offs in Ontologies

These aforementioned roles define the requirements we are after in finding ontologies suitable for our task. Ontologies can be found in numerous areas of computer science and information technology research ranging from information retrieval and knowledge management to artificial intelligence and machine learning. However, for the purpose of modelling the domain of existing web based systems, and more specifically for providing a basis for the corresponding user model, we are interested a few particular properties described below, along with a comparison of a few ontology representations in Table 1.

Table 1. Sample of Ontology Representations

	Ontology Representation				
	SUO KIF [5]	**OWL Family [6]**	**CycL [7]**	**Mecureo [8]**	**SMES [9]**
Engineering Formality	Manually, frame based	Manually, description logic	Manually, first order logic based	Automatic, lexical based	Automatic, lexical based
Serialisation	KIF, XML family	XML family	CycL, XML family	XML family	XML family
Available Tools	Limited	Many for construction and maintenance	Limited	Limited, but can use tools with OWL support	Limited
Reasoning Power	High	High	High	Limited, numerical	Limited, lexical
Validation	Strong	Moderate	Strong	Weak	Weak
Scrutability	Weak	Weak	Weak	Strong	Moderate

Engineering Formality

Semantic web based applications require a way to construct domain ontologies cheaply so they can be built in a reasonably fast manner [9].

Ushold and Gruninger describe different approaches to create ontologies in [2], being top-down, bottom-up and middle-out. The path taken is dependant on the available resources with which to construct the ontology. It is possible to start with an existing upper level ontology such as IEEE's Standard Upper Merged Ontology (SUMO) [5] and work top down from there. The SUMO website has references to numerous works extending the upper level ontology [1]. These formal ontologies often require the use of a domain expert and/or someone knowledgeable in the particular ontology philosophies and methodologies. A user may require guidance in understanding the underlying structure and axioms in formal ontologies. Having access to an ontologist or domain experts is rare, though additional tools or documentation will help users understand the reasoning and ontology.

On the opposite end of the spectrum, there are tools available to construct ontologies automatically from existing content (usually text-based such as domain documents, glossaries or dictionaries) which is often freely or cheaply available. Examples of work utilizing an automatic approach to creating ontologies include [9-11]. These automatic approaches will quite often involve a manual component anyway for light verification of the consistency.

Serialisation

Standardisation allows the use of existing ontology tools and resources. Standardisation also promotes reuse and extensibility. The semantic web initiative has intro-

[1] http://www.ontologyportal.org/Pubs.html

duced XML based serialization of ontologies ranging from using RDF [12] for simple and light-weight ontologies to OWL [6] for heavier-weight modelling. The IEEE Topic Map [13] standard provides another standard that would be appropriate for light weight ontologies. Table 1 suggests that it is fairly commonplace to include a XML based serialisation of the ontology (whether it be base XML, RDF, or OWL).

The specification and implementation should also be flexible enough to allow the ontology to be adapted to specialised domains and situations. For example, if the ontology is serialized to an XML format, will reasoning still work if the XML data is fragmented into smaller segments or sub-domains?

Reasoning Power

A prominent aspect of web based systems is the fact that the majority of content represents lower level concepts in the domain [14]. Therefore the ontology must support the user model in being able to reason about higher level concepts where there is little or no direct evidence. It must support reasoning about concepts that are not explicitly defined in the metadata set. This is especially important in student modelling systems where higher level concepts represent core learning goals of the student.

There are many ways we can reason about the domain we have modelled using the ontology. The structure and representation method of the ontology itself plays a large role in the implementation performance, accuracy and gestalt acceptance of the reasoning. In research areas such as information retrieval it is sufficient to use lexical ontologies to model users in order to expand on search queries [15]. We are interested to see what reasoning strategies would be appropriate for ontologies used in user modelling. In classic philosophy and artificial intelligence, there are two main methods of reasoning that can be employed, being *deductive* and *inductive*.

In [16], deduction or deductive arguments are defined as ones that given a set of premises that are true, then the conclusion must also be true. Using deductive reasoning in ontologies, new facts about the domain can be generated that must be conclusively true because the premises or statements that exist in the ontology are true. This relies on a representation of the ontology that has a high degree of formality that can enable deductive reasoning. It also requires a relatively complete model of the domain that includes higher level, fundamental concepts. Early designs in expert systems used representations such as predicate calculus and description logics that have well defined rules for stating propositions and making deduction. The Semantic Web ontology language, OWL [6], represents ontologies through description logic. However, the logical reasoning layer that sits on top of the representation is still a young area for research and development.

In contrast to deduction, inductive logic does not provide an assertion of truth to the conclusion. It provides a way to deduce that the conclusion is *probably* true based on the evidence supporting the premises [16]. Classical examples use Bayesian probabilities to illustrate inductive logic, and there are many existing systems that use this method to reason about users [17, 18]. The inherently evidence-based reasoning of user modelling makes inductive logic a more natural way to make inferences about users, and an overlaying method of inductive logic can be applied over even very simple light-weight ontologies.

Validation

How do we know the ontology provides a *correct representation* of the domain? Obviously, the *correct representation* is something that can only be judged by the people who use the ontology. So therefore there should be methods available to ensure that the ontology is easily validated for consistency. In ontologies with groundings in logic this can be done during the construction process when following formal methodologies [2]. The methodologies often employ the use of domain experts during the development process to craft the knowledge manually.

On the other hand, automatic and semi-automatically created ontologies tend to be of lower quality than manually created approaches in the sense that the representation is less formal and are often not backed by a strong upper level defining fundamental concepts.

Scrutability

In the spirit of supporting privacy legislation for electronic mediums [19], there is growing interest in allowing users to have an active role in the way their personal data is used. One way to achieve this is to let users directly interact and inspect their user model possibly correcting or removing data they feel is incorrect or inappropriate. In this instance, it is important for the user model to remain not only open but also scrutable [20]. In contrast, the idea of having open, scrutable user models of the learners in intelligent tutoring systems may even aid in reflection and provide additional educational benefits to both the instructor and the student [21, 22]. Because the ontology provides a foundational link between the user model and the domain content, it aids the user's comprehension of their user model and the domain itself.

Therefore ontology engineering methodology must support scrutability. This is very important if the scrutability of the ontology plays a role in the scrutability of the user model. When the ontology itself is based on structured domain content, as is often the case for (semi)automatically constructed ontologies [10, 23], the users can always refer back to these sources to get an understanding of the conceptualization.

Heavier-weight approaches may lose scrutability when users without a background in ontology engineering wish to examine the reasoning behind relationships and axioms. This is especially true in systems where domain experts and ontologists are involved only in the construction phase. At the same time, heavier-weight ontologies have a much higher cost in engineering effort as methodologies have to be adhered to.

4 Experiences in Ontology-based User Modelling

Our experience in ontology-based user modelling is in the context of teaching a course in User Interface Design and Programming. Students log onto a website to access learning objects consisting of slides with audio narration. We shall now describe the components in the system correlating to the architecture diagram in Figure 1.

We use the automatic ontology generation tool Mecureo [8] to generate an ontology of the HCI domain from an online glossary of HCI terms[2]. We have developed a metadata annotation tool called Metasaur [24] to aid in adding metadata to the learning objects and tutorials. Extensions to Metasaur allow us to also add terms to the glossary as we discovered the source glossary omitted a lot of core concepts.

For the user modelling server, the website utilises Personis [25]. Terms in the metadata set used to define components in the user model. So in effect, each page mapped to a number of components in the user model. As users accessed the page and listed to the audio, the system would add evidence to the terms in the user model that appeared on the metadata for that page. For example, a page that had terms *Cognitive Walkthrough* and *GOMS* would contribute evidence to the corresponding components in the user model. The values for the evidence were the length of time spent on the learning objects as well as the marks they receive in weekly tutorials.

We observe that the metadata term set only contains leaf concepts, and that evidence will mostly tend to feed into these leaf concepts rather than higher level concepts. For the user model interface we have developed a tool called called Scrutable Inference Viewer (SIV) [24]. SIV visualizes large use models using an innovative perspective distortion on font sizes to represent concepts and their related concepts. It represents the values for the concepts through the font colour. We have implemented a mechanism in SIV that reasons about concepts at the higher level where there is little or no evidence. We have implemented a simple algorithm similar to spreading activation where resolved values for concepts can contribute to the values of neighbouring concepts. This way we can recursively apply this algorithm up from leaf to higher level concepts to infer a value about how well the student understands that concept.

5 Discussion

Because the content becomes evidence sources to the user model, there is a requirement that it shares an understanding of the domain with the user modelling system. Therefore the ontology must fulfill a number of roles. It defines a set vocabulary that enables metadata annotation of the content. Secondly, it provides a mechanism for reasoning about the users, especially in higher level concepts that may not directly appear in the metadata. And finally, provides support scrutability, for aiding the user in getting a better understanding of the domain, the adaptations and themselves if they wish to examine their user model. We have also stated a number of desirable properties for ontologies and included references to existing ontologies, as well as discuss trade-offs and dichotomies within them.

Where traditional approaches have favoured more rigorous and formal ontologies, our own experience shows that there are advantages in using lighter-weight approaches. Reasoning about concepts could be as simple as spanning out to neighbours to find related concepts to a more mathematical system employing algorithms similar to those found in spreading activation research.

[2] http://www.usabilityfirst.com/glossary/

We conclude with a quote from Sparck [26], "Overall the trend in document characterisation has been away from lexical normalisation and towards relational simplification, i.e. towards decreasing ontological expressiveness, decreasing epistemological commitment, and decreasing inferential power. But this has been correlated with wider application and better task performance".

In or own work we utilize the constructing ontologies automatically, and although there are trade-offs in formality and consistency, the manual time spent in engineering is greatly reduced, and users can be referenced back to the original information sources that the ontology is based on when scrutinizing the adaptations. We may find that simpler inductive inference algorithms will suffice for reasoning about users in adaptive systems.

References

We thank Hewlett-Packard for funding this research.

References

1. Gruber, T.: Toward Principles for the Design of Ontologies Used for Knowledge Sharing. Formal Ontology in Conceptual Analysis and Knowledge Representation. (1993)
2. Uschold, M. and M. Gruninger: ONTOLOGIES: Principles, Methods and Applications. Knowledge Engineering Review. 11 (2) (1996)
3. Fensel, D.: Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce, Springer. (2001).
4. Berners-Lee, T., J. Hendler, and O. Lassila, The Semantic Web, in Scientific America. 2001.
5. Niles, I. and A. Pease: Towards a Standard Upper Ontology. in 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001). Ogunquit, Maine (2001)
6. W3C: OWL Web Ontology Language Overview. (2003) Available from: http://www.w3.org/TR/owl-features/
7. Lenat, D. and V. Guha: Building large knowledge-based systems. Representation and inference in the Cyc project., Addison-Wesley Reading, Massachusetts, USA. (1990).
8. Apted, T. and J. Kay: MECUREO Ontology and Modelling Tools Special Issue on Concepts and Ontologies. WBES of the International Journal of Continuing Engineering Education and Lifelong Learning. (Accepted 2003, to appear)
9. Maedche, A. and S. Staab: Ontology Learning for the Semantic Web. IEEE Intelligent Systems. 16 (2) (2001) 72-79
10. Reimer, U., et al.: Ontology-based Knowledge Management at Work: The Swiss Life Case Studies, in Towards the Semantic Web: Ontology-driven Knowledge Management, J. Davis, D. Fensel, and F.v. Harmelen, Editors, John Wiley & Sons: West Sussex, England (2003) 197-218
11. Sabou, M.: Extracting Ontologies from Software Documentation: a Semi-Automatic Method and its Evaluation. in ECAI-2004 Workshop on Ontology Learning and Population (ECAI-OLP). Valencia, Spain (2004)
12. Brickley, D. and R.V. Guha: RDF Vocabulary Description Language 1.0: RDF Schema. (2003) Available from: http://www.w3.org/TR/rdf-schema/

13. IEEE: XML Topic Maps (XTM) 1.0. (2001) Available from: http://www.topicmaps.org/xml/1.0/
14. De Bra, P. and L. Calvi: AHA: a Generic Adaptive Hypermedia System. in Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia HYPERTEXT'98. Pittsburgh, USA (1998) 20-24
15. Luca, E.W.D. and A. Nurnberger: The Importance of Ontology Based User Modeling and Adaptation in Information Retrieval. (2004)
16. Copi, I.M.: Introduction to Logic. 3rd ed, MacMillan Company New York. (1968).
17. Conati, C., A. Gertner, and K. Vanlehn: Using Bayesian Networks to Manage Uncertainty in Student Modeling. User Modeling and User-Adapted Interaction. 12 (2002) 371-417
18. Zapata-Rivera, J.D. and J. Greer: Inspecting and Visualizing Distributed Bayesian Student Models, in Intelligent Tutoring Systems, G. Gauthier, C. Frasson, and K. VanLehn, Editors (2000) 544-553
19. Kobsa, A.: Personalised Hypermedia and International Privacy. Communications of the ACM. (2002)
20. Kay, J., A scrutable user modelling shell for user-adapted interaction, in Basser Department of Computer Science. 1999, University of Sydney: Sydney, Australia.
21. Kay, J.: Learner control. User Modeling and User-Adapted Interaction. 11 (1-2) (2001) 111-127
22. Bull, S. and H. Pain: 'Did I Say What I Think I Said, And Do You Agree With Me?' Inspecting and Questioning the Student Model. in World Conference on Artificial Intelligence and Education, AACE. Charlottesville VA (1995) 501-508
23. Sabou, M.: Extracting Ontologies from Software Documentation: a Semi-Automatic Method and its Evaluation. in ECAI-2004 Workshop on Ontology Learning and Population (ECAI-OLP). Valencia, Spain (2004)
24. Kay, J. and A. Lum: Building user models from observations of users accessing multimedia learning objects, in Proceedings of Adaptive Multimedia Retrieval, N. A and N. Detyniecki, Editors, Springer (2004) 36–57
25. Kay, J., B. Kummerfeld, and P. Lauder: Personis: a server for user models. in Adaptive Hypertext: Springer (2002) 203-212
26. Jones, K.S.: What's new about the Semantic Web? some questions. ACM SIGIR Forum archive. 38 (2) (2004) 18-23