

Exploiting Tagging in Ontology-based e-Learning

Nicola Capuano¹, Angelo Gaeta¹, Francesco Orciuoli¹ and Stefano Paolozzi²

¹ Dipartimento di Ingegneria dell'Informazione e Matematica Applicata, University of Salerno
Via Ponte Don Melillo, 84084 Fisciano, Salerno, Italy

² Dipartimento di Informatica e Automazione, University of Roma Tre
Via della Vasca Navale, 79, I, 00146 Roma, Italy

Abstract. The ontologies are used to state the meaning of the terms used in data produced, shared and consumed within the context of Semantic Web applications. The folksonomies instead are an emergent phenomenon of the Social Web and represent the result of free tagging of information and objects in a social environment. Both ontologies and folksonomies are considered useful mechanisms to manage the information and are pretty always exploited, independently, in several areas of interest in order to cope with different problems related to searching, filtering, categorization and organization of content within some applications for e-commerce, e-learning, e-science, etc. In our opinion the two mechanisms are not in opposition but could be synergically used. In this paper we propose an approach based on the convergence between ontologies and folksonomies in order to improve personalised e-learning processes.

1 Introduction

In a famous presentation [6], T. Berners-Lee explains his vision of the Semantic Web. He believes that the Web of the future foresees machines able to understand the content of web pages. The core of his Semantic Web architecture is composed of three main layers. At the bottom there are the Markup Languages for the resources description. On top of this layer he places the ontologies used to define terms and their relationships between other terms. Ontologies enable the next layer, namely the Logic Layer, where we can deduce new information by analyzing assertions from the Web. Hence, ontologies are considered an important building block for the evolutions of the Web. Gruber, in [7], states that an ontology is an explicit specification of a conceptualization and, more pragmatically, he explains that a common ontology defines the vocabulary by which queries and assertions are exchanged among agents (both human and software agents).

Let us to illustrate two important aspects related to the use of ontologies: (i) the fields of application in which ontologies should be used and (ii) the approaches to ontologies definition on the Web.

With respect to the first issue, in [8] it is well described that the ontologies can provide guidance on how to correctly relate pieces of information in a specific domain, can provide a more effective basis for information extraction or content clustering, may

be a source of structure and controlled vocabularies helpful for disambiguating context, can provide guiding structure for browsing or discovery within a domain, can provide the basis to reason or infer over its domain and can be organized hierarchically from the most specific to the most abstract one.

Moreover, concerning the approaches to ontologies on the Web, both structure and formalism are dimensions for classifying them, which combined are often referred to as an ontology expressiveness. Some interesting ontology approaches are:

- **Hierarchical Faceted Metadata.** A faceted classification [9] allows the assignment of more than one classification to an object, enabling the classifications to be organized in multiple ways, rather than in a unique taxonomic order. For instance, a collection of photos might be classified using an author facet, a subject facet, a date facet, etc. Faceted classification supports the navigation of information along multiple paths corresponding to different orderings of the facets.
- **Folksonomies.** A folksonomy [10] is the result of free tagging of content (anything with a URL) for the aims of simplifying successive searching operations. When the tagging is done in a social environment (e.g. Flickr, Del.icio.us, Furl, etc.), a folksonomy is created from the act of tagging by the users consuming the information. The value in this external tagging is derived from people using their own vocabulary and adding explicit meaning, which may come from inferred understanding of the tagged information/content.
- **Topic Maps.** A Topic Map³ represents information using topics (i.e. any concept or subject coming from a specific domain), associations (relationships between topics) and occurrences (relationships between topics and resources/content relevant to them).
- **OWL Lite, DL, Full.** The Web Ontology Language (OWL)⁴ is a language (based on XML and RDF/RDF-S) used for defining ontologies for the Web. An OWL ontology includes descriptions of classes, properties and instances and is designed for use by applications that need to process the information content.
- **Higher-order Formal and Upper-level Ontologies.** SUMO⁵, DOLCE⁶ and OpenCyc⁷ are examples of these ontologies that have the goal of enabling AI applications to perform human-like reasoning.

More formal ontologies (e.g. OWL-Full, OWL-DL, Higher-order formal and upper-level ontologies, etc.) have greater expressiveness, structure and inferential power than the less formal ones (e.g. folksonomies, Topic Maps, Hierarchical Faceted Metadata, etc.). The higher the formality, the higher the effort and rigor required, resulting in an approach that is more powerful but also more rigid and less flexible. Furthermore, formal ontologies require strong knowledge representation skills that are not simply findable. These are all aspects that should be taken into consideration when choosing an ontology approach. In particular, approaches like folksonomy are fine when ontologies producers are typical Web 2.0 users and the expected results are to categorize

³ <http://www.ontopia.net/topicmaps/materials/tao.html>

⁴ <http://www.w3.org/2004/OWL/>

⁵ <http://www.ontologyportal.org/>

⁶ <http://www.loa-cnr.it/DOLCE.html>

⁷ <http://www.cyc.com/cyc/opencyc/overview>

content and support search operations. Otherwise, the adoption of Topic Maps is preferable when we aim to model educational domains (e.g. Physics, Mathematics, etc.), the ontologies producers are domain experts (e.g. professors) having no Knowledge Engineering skills and there are domain-specific inference rules to be executed. In the end, we could choose OWL if we need a standard reasoning like, for instance, checking if a concept C subsumes the concept D or if an individual a is an instance of concept F . *Is there a way to combine some of them in order to make the ontologies construction process more rapid and simpler, and the ontologies management processes more effective and efficient?*

The aim of this work is to answer the previous question illustrating the existing works and proposing an approach to face and solve the problem in the e-learning area, making specific reference to an existing e-Learning Platform. In the section 2 we will illustrate some approaches for the convergence of more formal ontologies and less formal ones and some specifications useful to manage the ontological structures. In section 3 we will present the Intelligent Web Teacher (IWT) [13], an e-Learning Platform enabling the definition and execution of personalized learning experiences based on the explicit representation of educational domains knowledge. In the section 4 we will propose an approach to improve the ontology-related processes in IWT. In the end, in section 5 we will provide considerations about this work and some ideas for future research activities related to it.

2 Background and Related Works

Semantic Web researchers have become increasingly interested in studying different ways to use the technologies of the Semantic Web to organize data coming from the Social Web. Some groups have studied possible means to improve the way the social tagging system is organized. Laniado [17] suggests using WordNet⁸ to improve the performance of the related tags search of Del.icio.us. A promising field of research has been the use of folksonomies as corpora to extract domain specific ontologies.

VanDamme [19] proposes an approach (named FolksOntology) that integrates multiple resources and techniques to achieve this goal. Some examples of techniques: compute the similarity degree between two tags by counting the number of times they are used to describe the same object, cluster the actors and the groups that share the same tags or the same objects. WordNet, Wikipedia⁹ and Leo Dictionary¹⁰ are used as resources to identify spelling errors in tags, to substitute tags with their hyponyms and to translate tags from one language into another.

Torniai [22] analyses this problem in the context of e-Learning: it is difficult to create and maintain the ontologies that describe different study courses. Torniai's group extended LOCO-Analyst, an e-learning ontology based learning tool, in order to exploit the students annotations to update domain ontologies. Students can associate tags to learning objects using OATS (Open Annotation and Tagging System). Teachers can see both the ontology graph and the tag cloud of the students annotations: when a teacher

⁸ <http://wordnet.princeton.edu/>

⁹ <http://www.wikipedia.org/>

¹⁰ <http://dict.leo.org/>

selects a lesson, the related tag cloud is visualized (the size of each tag depends on its popularity) and the related concepts are highlighted in the graph. If a single concept in the graph is selected, related tags are colored with a darker shade of blue.

The folksonomies, anyway, have not been considered only as a corpus of information for data mining processes. In recent times the Semantic Web community has noticed that social web sites behave like data silos and that information cannot be exchanged between them in an easy manner. There is a need to improve sharing and interoperability. Gruber was one of the firsts to propose the creation of an ontology of folksonomies [16]. He sees them as a substrate to be used to interface with different social tagging systems and share tags between them improving the classification and helping research work. According to Gruber, the tagging relationship should be expressed as a tuple (object, tag, tagger, source, + or -), where the source is the social tagging system from which the tag is taken and the fifth field is a polarity argument (plus for positive tagging, minus to signal the tag as spam). Other researchers tried to use variations of his tagging relation to construct a tag ontology, sometimes using existing ontologies as building blocks. According to Knerr [20] the folksonomy system architecture should be changed in order to record in one place a users personal information and relations, coupled with the tags they entered in all the folksonomies they are part of. Knerr proposes a Tag Ontology (known as TagOnt), in which Tagging, the central element of the ontology, is expressed as a tuple (*time, user, domain, visibility, tag, resource, type*). This definition is similar to that of [16] (with domain corresponding to Grubers source) but it adds the parameter *time* that keeps track of the tagging date.

The existing ontologies are used to map individual parts of an ontology: Dublin Core¹¹ for time, FOAF¹² for users, SKOS¹³ for tags, RDFS¹⁴ for resources. The visibility parameter can take the values public, private, or protected. Newman has created another Tag Ontology, which is widely used in the Semantic Web field [18]. The tagging relation is expressed by the tuple (*User, Resource, Tag*). Every single tag is identified with an URI, allowing easier research and clustering. The tag relations are modeled using SKOS; even the Tag class inherits from the SKOS Concept Class. Even if it is widely used, the Tag Ontology has suffered some criticism because of the lack of semantics in tag relationships: we cant specify if two words are related because they are synonyms or because they are variations of the same word. One of the most important ontologies used to exchange data taken from social communities is SIOC (Semantically-Interlinked Online Communities), described in detail in [1]. With SIOC is possible to keep track of the main argument of a discussion, the number of page views, the author and the relationships between users.

Moreover, the MOAT project [21] is a framework created by Alexandre Passant and composed of an ontology, a client and a server. The MOAT ontology differs from previous tagging ontologies because the tagging relationship includes a new concept: meaning. The Tag Ontology Class is extended: each tag has a set of meanings, described with an URI (usually taken from lexical resources like DBpedia) and the set of users

¹¹ <http://dublincore.org/documents/dces/>

¹² <http://www.foaf-project.org/>

¹³ <http://www.w3.org/2004/02/skos/>

¹⁴ <http://www.w3.org/TR/rdf-schema/>

that associated a certain meaning to a given tag. The MOAT client can be added to every blog application and offers an interface to associate meanings (in URI format) to tags, while users who subscribe to the MOAT server can share tags and meanings with other people.

In the end, SCOT (Social Semantic Cloud of Tags) is an ontology created with the purpose of enhancing tag sharing and interoperability among different social communities [18]. The tagging activity is represented as a ternary relation between users, tags and resources. The SCOT model is based on three existing ontologies: like TagOnt, it reuses FOAF and SKOS for mapping users and tags, but uses SIOC to represent resources. Int.ere.st¹⁵ is a social tagging, bookmarking and sharing service based on the SCOT ontology [18]. A more complete coverage of these needs can be achieved by federating these specifications and exploit the advantages of each one of them.

3 Ontological Structures in the Intelligent Web Teacher

The Intelligent Web Teacher (IWT) is primarily an e-Learning Platform that enables the definition and the execution of personalized learning experiences packaged in a Unit of Learning (i.e. a course, a module or a lesson structured as a sequence of Learning Activities represented by Learning Objects and/or Learning Services). The foundation for the Unit of Learning (UoL) building process is the Learning Model described in [11]. The Learning Model allows to automatically generate a UoL and to dynamically adapt it during the learning process according to the learners' preferences and cognitive state (personalization process). In order to achieve the expected adaptation capability, the Learning Model uses three specific sub-models: the Knowledge Model, the Learner Model and the Didactic Model, which are exploited by a specific process used to define personalized e-learning experiences.

The Knowledge Model describes, in a machine-understandable way, the subset of the educational domain that is relevant for the e-learning experience we want to define, concretize and broadcast. The Knowledge Model exploits the ontologies. The used ontology approach is really similar to that of Topic Maps introduced in section 1 and is described in [14]. In our approach the vocabularies are composed of terms representing subjects that are relevant for the educational domain we want to model. Subjects are associated to other subjects through a set of several conceptual relations. The most important relations are: *HasPart* (in brief HP) that is a part-of relation and *IsRequiredBy* (in brief IRB) that is an order relation. The ontologies constructed following the quite a few aforementioned informal rules are called e-learning ontologies. Observe that when we refer to concepts in the e-learning ontologies we are referring to the subjects of the educational domain we are modeling. Let us now consider how to build an e-learning ontology. Suppose we have to model the educational domain D , so we try to conceptualize the knowledge underlying D and find a set of terms representing relevant concepts in it. The result of the previous step is the list of terms $T = C, C_1, C_2, C_3$ where T is one of the plausible conceptualizations of D . The existence of relations $HasPart(C, C_1)$, $HasPart(C, C_2)$ and $HasPart(C, C_3)$ means that in order to learn a subject C the

¹⁵ <http://int.ere.st/>

learners have to learn subjects C_1 , C_2 and C_3 without considering a specific order. If we add the relations $IsRequiredBy(C_1, C_2)$ and $IsRequiredBy(C_3, C_2)$ to the previous set of relations we can state that C_1 has to be necessarily learned before C_2 and C_3 has to be necessarily learned before C_2 .

Now, we would like to introduce new elements called Learning Objects. You can interpret the connection between a concept and a Learning Object, for instance C_1 and LO_1 , as a *HasResource* (in brief HR) relation. The relation $HasResource(C_x, LO_y)$ means that the educational content packaged in Learning Object LO_y explains concept C_x . So, if we assume that $HasResource(C_1, LO_1)$, $HasResource(C_2, LO_2)$, $HasResource(C_3, LO_3)$ and that our Learning Objective is C_1 then the corresponding assembled e-learning experience is composed only by $[LO_1]$, otherwise if the Learning Objective is C then the assembled e-learning experience will be composed as $[LO_1, LO_3, LO_2]$.

For sake of simplicity, we disregard the Didactic Model (used to model learning methods to be applied to specific learning experiences) and the Learner Model (used to model cognitive states and learning preferences of single learners in order to support the personalization process) given that it is not fundamental for the focus of this work. Excluding the selection of the Learning Objective over an ontology and some other customization parameters, the UoL building process is fully automatic and realized through the execution of several algorithms. The most important are: *Learning Path Generation Algorithm* and *Presentation Generation Algorithm* [12]. Using these algorithms it is possible to generate courses tailored to a class, to a specific group and even to single learners.

According to the IWT approach, a Learning Object is a learning content (or a packaged aggregation of learning content) that can be delivered through a Web Browser, that is annotated with an instance of a metadata schema interoperable with IEEE LOM [15] and that is stored and indexed into a Learning Object Repository. The binding between Learning Objects and subjects of ontologies is realized by storing subjects references in a specific attribute (*Classification.Taxonpath.Taxon* [15]) within the metadata instances associated with Learning Objects. The binding operation is performed by users (with resource manager permissions) selecting the object, the metadata attribute, the ontology of interest and one or more subjects upon the selected ontology.

In IWT, the Learning Objects, and in general all content managed by the Platform, can be tagged both in a personal and in a shared area. The tagging process is free (no guidance for tags definition). In the context of the e-Learning Platform, the tagging process is used in order to improve the searching functionality, by using the Tag Cloud facility, and to support collaborative learning activities where the teacher asks for the students to collaboratively categorize a set of resources as a part of a whole learning scenario. The assigned tags are stored into the *General.Keyword* [15] Learning Object metadata attribute and used to create indexes for performance aims. The result of the tagging process is a folksonomy that grows when new learning content are added and represents a rich source of knowledge that could be better exploited with respect to the optimization of the e-Learning Platform processes.

The mechanisms concerned with the e-learning ontologies and the content tagging are independently managed and controlled by the IWT Platform services. This independence leads to:

- Possible redundancy of labels applied to content. The same Learning Object could be annotated at the same time with the tag "SOA" and with a reference to a subject "SOA" within the "Software Architectures" ontology.
- Possible disorientation for users. The users could be confused if involved in different, but similar, tagging operations to perform in different contexts and with different goals.
- Difficulty in sharing, within the users community, the intended meaning of tags, keywords, labels, etc. From the point of view of handling unambiguous meanings, a unique space of shared terms is easier to manage than multiple spaces .
- Unnecessary complexity of the search engine. Having more than one indexing mechanism triggers the growth in complexity of the search engine that has to query several indexes and harmonize the result sets.
- Incapability to exploit tags within the Unit of Learning building process. The aforementioned Presentation Generation Algorithm retrieves Learning Objects using only subjects previously extracted from the reference ontology and cannot exploit other semantic enrichments provided by users in order to improve the filtering action of the algorithm.

4 A Proposed Approach: The Tagging System

In order to face the problems illustrated in section 3 we propose an approach based on the unification of the e-learning ontologies management and the content tagging mechanisms within the IWT Platform.

The identified solution is focused on the action of providing a unique tag space populated by several content and knowledge managing processes. The conceptual architecture of the Tagging System is reported in figure 1(a).

So, with respect to the creation of e-learning ontologies, the subjects (nodes in the ontologies) will be retrieved from tags stored in the tag space. The link between ontologies and Learning Objects (defined as the *HasResource* in section 3) is realized only using the *General.Keyword* attribute of the Learning Object Metadata schema, that is also used to store simple tagging information. The *Classification.Taxonpath.Taxon* will no longer be used. In figure 1(b) we show the use of Tag Ontology, SCOT and MOAT to implement respectively the tag modeling, the e-learning ontologies structure representation and the synonyms handling with respect to the proposed approach.

Numerous are the interaction flows involving users, tag space and other IWT modules. We can deal with the aforementioned flows by decomposing them into classes: *Insert*, *Import*, *Search*, *Organize*, *Suggest* and *Export* flows.

The *Insert* flows concern with tag space population. The new subjects defined by the authors of ontologies will become tags in the tag space, everytime the users add a keyword to a given content (e.g. Learning Objects), this keyword is inserted into the tag space as a new tag. Furthermore, new tags used in the Blog, Wiki, Forum, etc. will be inserted as tags in the tag space.

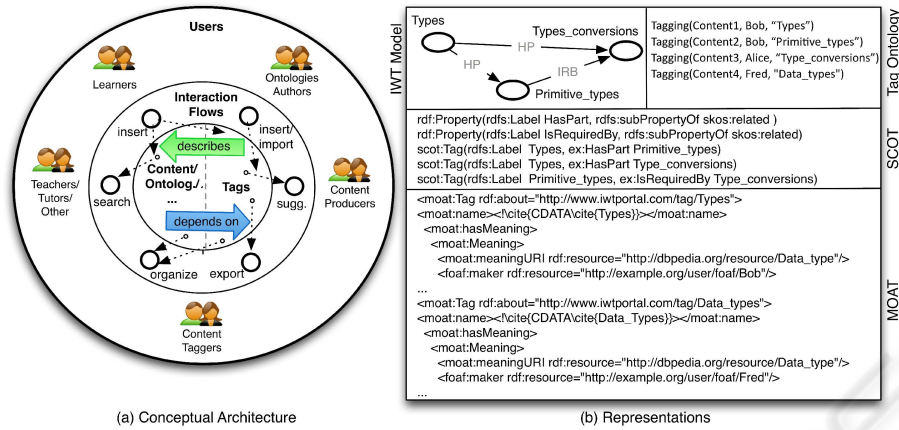


Fig. 1. Tagging System.

The *Import* and the *Export* flows concern the interoperability with external tag spaces. With respect to the interoperability issues, the SCOT specifications represents a good solution.

The *Search* flows regard the way users can exploit the tags associated with the content. Two are the main search mechanisms taking advantage of the tag space. The first one is the keyword-based search and the second one is the Tag Clouds [5], i.e. visual representations of social tags, displayed in paragraph-style layout, usually in alphabetical order, where the relative size and weight of the font for each tag corresponds to the relative frequency of its use.

The *Suggest* flows are involved in exploiting specific elaboration performed on the tag space in order to support the users when they use tags. Suggestions about tags already existing in the tag space can be useful to avoid the occurrence of different tags with the same implicit meaning. This suggestion could be provided by the system using approaches like [4] where a technique is proposed to compute semantic relatedness between two terms using Wikipedia or like [3] where the authors define the Google distance as a measure of semantic relatedness computed by counting the number of hits that two keywords have if used together in the Google search engine.

The *Organize* flows concern a way of supporting users who build e-Learning Ontologies by providing draft structures (to be manually refined) automatically extracted from the tag space. Works like [2] and [19] are interesting for this goal. In particular in [2], the authors aim at adding meaning to tags, using generic ontologies (WordNet) and domain specific ontologies. Domain specific ontologies could be created by analysing set of Flickr pictures related to the same subject, the most common tags are extracted and used to individualise the basic concepts of the ontology.

5 Conclusions and Future Works

In the present work we have proposed (section 4) an approach with the objective to face and solve the problems identified at the end of the section 3. In particular:

- The redundancy of labels applied to the content is loosened by the adoption of a unique tag space.
- The disorientation for users is loosened by the unique way to tag content and by the use of tags as ontologies subjects and as keywords.
- The difficulty in sharing, within the users community, the intended meaning of tags, keywords, labels, etc is loosened by the adoption of specifications like MOAT and SCOT and by the model provided by the Tag Ontology.
- The complexity of the search engine is loosened by the adoption of a model in which content are described only by tags. Advanced searches are provided by means of similarity algorithms and other tag space elaborations.
- The incapability to exploit tags within the Unit of Learning building process is loosened by modifying the Presentation Generation Algorithm in order to retrieves Learning Objects exploiting tags stored in *General.Keyword* field of metadata schema and used as subjects within the e-Learning Ontologies.

Future works will concern the analysis, design and prototyping of a system based on the proposed approach and integrated with the IWT Platform in order to perform experiments and evaluate our approach.

References

1. Breslin, J.G., Harth, A., Bojars, U., Decker, S.: Towards semantically-interlinked online communities. ESWC 2005. Heraklion, Crete, Greece, May 29 - June 1, 2005, Proceedings. (2005).
2. Lee, S.S. , Yong, H.S.: OntoSonomy: Ontology-based Extension of Folksonomy doi.ieeecomputersociety.org (2008)
3. Cilibrasi, R., Vitanyi, P.: The Google Similarity Distance. IEEE Transactions on knowledge and data engineering 19(3) (2007) 370383.
4. Strube, M., Ponzetto, S.P.: WikiRelate! Computing semantic relatedness using Wikipedia. In Proc. of AAAI-06, pp. 14191424. (2006).
5. Hearst, M. A., Rosner, D.K.: Tag Clouds: Data Analysis Tool or Social Signaller? HICSS 2008. <http://people.ischool.berkeley.edu/hearst/papers/tagclouds.pdf>.
6. Berners-Lee, T.: Semantic Web on XML, Keynote presentation for XML 2000. Slides available at: <http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide1-0.html>. Reporting available at: <http://www.xml.com/pub/a/2000/12/xml2000/timbl.html> .
7. Gruber, T. R.: Towards Principles for the Design of Ontologies used for Knowledge Sharing. International Journal of Human-Computer Studies, 43:907-928, 1995.
8. Bergman, M. K.: An Intrepid Guide to Ontologies. Theres an Endless Variety of World Views, and Almost as Many Ways to Organize and Describe Them. AI3::Adaptive 2007. Retrieved from: <http://www.mkbergman.com/?p=374>
9. Denton, William.: How to Make a Faceted Classification and Put It On the Web. Nov. 2003. <http://www.miskatonic.org/library/facet-web>.
10. Vanderwal, T.: Folksonomy Coinage and Definition. <http://www.vanderwal.net/folksonomy.html>, 2004.
11. Albano, G., Gaeta, M., Salerno, S.: E-learning: a model and process proposal. IJKL, vol. 2, no.1, 2006, pp. 73-88.
12. Capuano, N. , Gaeta, M., Miranda, S. , Orciuoli, F. and Ritrovato, P.: LIA: An Intelligent Advisor for e-Learning. 1st world summit on The Knowledge Society, 2008.

13. Albano, G., Gaeta, M., Ritrovato, P.: IWT: an innovative solution for AGS e-Learning model. *IJKL*, vol. 3, n. 2-3, 2007.
14. Gaeta, M., Orciuoli, F., Ritrovato, P.: Advanced ontology management system for personalised e-Learning, *Knowl. Based Syst.* (2009), doi: 10.1016/j.knosys.2009.01.006
15. Learning Technology Standards Committee of the IEEE: Draft Standard for Learning Objects Metadata, IEEE Draft P1484.12.1/D6.412, 2002 http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf.
16. Gruber, T. R.: Ontology of Folksonomy: A Mash-up of Apples and Oranges. Invited keynote, November 2005.
17. Laniado, D., Eynard, D. and M. Colombetti: Using WordNet to turn a folksonomy into a hierarchy of concepts. In *Proc.of Semantic Web Application and Perspectives. 4th Italian Semantic Web Workshop*, pages 192-201, Bari, Italy, 2007.
18. Kim, H.L., Breslin, G.J., Yang, S.K., and Kim, H.G.: *Int.ere.st: Building a Tag Sharing Service with the SCOT Ontology*. In *Proceedings of the AAAI 2008 Spring Symposium on Social Information Processing*, Stanford University, California, 2008.
19. Van Damme, C., Hepp, M. and Siorpaes, K.: *FolksOntology: An Integrated Approach for Turning Folksonomies into Ontologies. Bridging the Gap between Semantic Web and Web 2.0 Workshop at the ESWC 2007*. 2007, Innsbruck, Austria.
20. Knerr, T.: *Tagging ontology- towards a common ontology for folksonomies*. (2006) Retrieved June 14, 2008, from <http://tagont.googlecode.com/files/TagOntPaper.pdf>.
21. Passant, A. and Laublet, P.: *Meaning Of A Tag: A collaborative approach to bridge the gap between tagging and Linked Data*. In *Proceedings of the WWW 2008 Workshop Linked Data on the Web (LDOW2008)* Beijing, China, Apr 2008.
22. Torniai, C., Jovanovic, J., Bateman, S., Gaevic, D., Marek, H.: *Leveraging Folksonomies for Ontology Evolution in E-learning Environments*. *The IEEE International Conference on Semantic Computing*, 2008.

