User ratings of ontologies: Who will rate the raters?

Natalya F. Noy
Stanford Medical Informatics
Stanford University
Stanford, CA
noy@smi.stanford.edu

Ramanathan Guha
IBM Almaden Research Center
San Jose, CA
rguha@ibm.com

Mark A. Musen
Stanford Medical Informatics
Stanford University
Stanford, CA
musen@smi.stanford.edu

Abstract

The number of ontologies and knowledge bases covering different domains and available on the World-Wide Web is steadily growing. As more ontologies are available, it is becoming harder, and not easier, for users to find ontologies they need. How do they evaluate if a particular ontology is appropriate for their task? How do they choose among many ontologies for the same domain? We argue that allowing users on the Web to annotate and review ontologies is an important step in facilitating ontology evaluation and reuse for others. However, opening the system to everyone on the Web poses a problem of trust: Users must be able to identify reviews and annotations that are useful for *them*. We discuss the kinds of metadata that we can collect from users and authors of ontologies in the form of annotations and reviews, explore the use of an Open Rating System for evaluating ontologies and knowledge sources, and present a brief overview of a Web-based browser for Protégé ontologies that enables users to annotate information in ontologies.

Ontologies On The Web Scale

The number of ontologies and knowledge bases covering different domains and available on the World-Wide Web is steadily growing. Ontologies constitute the backbone of the Semantic Web and their number is steadily growing. The Swoogle crawler,[1] for example, indexes more than 4000 ontologies at the time of this writing.

It is commonly agreed that one of the reasons ontologies became popular is because they hold a promise of facilitating interoperation between software resources by virtue of being shared agreed-upon descriptions of domains used by different agents. Such interoperation is, for example, a key requirement for the Semantic Web to succeed. Suppose we are developing a Semantic Web service that uses an ontology. If we choose to reuse an existing ontology to support our service rather than to create a new one, we get the interoperation with the others using the same ontology "for free." In addition, we save time and money required to develop an ontology and we get the benefit of using an ontology that has already been tested by others.

However, as more ontologies become available, it becomes harder, rather than easier, to find an ontology to reuse for a particular application or task. Even today (and the situation will get only worse), it is often easier to develop a new ontology from scratch than to reuse someone else's ontology that is already available. First, ontologies and other knowledge sources *vary widely* in quality, coverage, level of detail, and so on. Second, in general, there are very few, if any, objective and computable measures to determine the quality of an ontology. Deciding whether an ontology is appropriate for particular use is a *subjective* task. We can often agree on what a bad ontology is, but most people would find it hard to agree on a universal "good" ontology: an ontology that is good for one task may not be appropriate for another. Third, while it would be helpful to know how a particular ontology was *used* and which applications found it appropriate, this information is almost never available.

We believe that having a large number of reviews and annotations generated both by ontology *authors* and *users* is the crucial component in enabling reuse of ontologies and other knowledge sources that have to be evaluated subjectively. The idea is not unlike rating consumer products or books: there is no perfect coffeemaker or perfect book for everyone, there is no uniform "best" measure in either category, and therefore we rely on reviews by others to help us decide what to buy. As with existing review and rating systems, such as Epinions[2] and Amazon,[3] the scale of the Web helps by providing a huge number of potential reviewers, but it also poses new challenges: It is inevitable that some significant portion of such reviews and annotations will be of low quality. Furthermore, evaluation and review of ontologies pose additional challenge: if our application needs a simple hierarchy of classes in a particular domain, an excellent review from a trusted user who prizes the quality and number of formal axioms over

[1] http://www.swoogle.org

[2] http://www.epinions.com

[3] http://www.amazon.com

coverage or simplicity would probably still count as an "unhelpful" review for us. Therefore, when deciding whom to trust, we must take into account which dimensions of evaluation we are interested in and who we trust in assessing this particular dimension.

In this paper, we discuss the kinds of metadata that we can collect from users and authors of ontologies, we explore the use of an Open Rating System for evaluating ontologies and knowledge sources, and present a brief overview of a Web-based browser for Protégé[4] ontologies that enables users to annotate information in ontologies. The prototype itself is very preliminary and the main contributions of the paper are identification of the type of metainformation that can be collected from users, and the concept of using an Open Rating System and Web of Trust for ontology evaluation.

Metadata for Describing Ontologies and Knowledge Sources

We envision that there will be a number of virtual repositories and directories that will collect information about ontologies available on the Semantic Web. The information in these directories will likely be collected and compiled by a combination of automatic tools, such as crawlers and tools for ontology analysis, and authors and users that submit ontologies to the repositories and provide metadata, annotations, and reviews of ontologies. In the early days of the Web, users wanted to have their home pages listed in directories such as Yahoo! and have often gone to the trouble of manually submitting the links to their pages to the more prominent directories. We can probably expect a similar phenomenon for the ontologies on the Semantic Web: At least initially, ontology developers would want to expend extra effort to facilitate the user of their ontologies by others.

There are three types of information about ontologies that can be available in any ontology repository: (1) information provided by ontology *authors*; (2) information collected *automatically*; and (3) information provided by *users* of the repository.

When authors submit an ontology to a repository or put metadata in its definition, we can usually expect them to provide some salient information about their ontologies. Arpírez and colleagues, for example, have developed a reference ontology for the type of information that the authors can provide (Arpírez *et al.* 2000). At the minimum, we can expect to find annotation on some of the following parameters:

- Domain of the ontology, informal description of the content: Many ontologies already come with the metadata that informally describes heir content. There may be a controlled vocabulary for indicating the domain of the ontology (for aggregating the information) and an option to add new domains.
- Purpose of the ontology, coverage: Is it general or specific, is it designed for particular kinds of applications?
- Availability of the ontology: Is it freely available? Is registration required? How can new versions be obtained?
- Ontology language and tools: Which ontology language is it available in? Which tools were used to develop it?

Aggregating and providing searches based on this information would be the first step in helping users find what they need. However, as we have discussed earlier, this information is not nearly enough to help a user sort out through all the ontologies covering the same domain.

A number of automated tools can help the user to understand what is available in the ontologies that match his initial search criteria. First, an automatically produced *summary* of an ontology can include, for example, top levels in the ontology's class hierarchy; a graphical representation of these top-level concepts and links between them. Tools can generate these top-level snapshots automatically or allow ontology authors to include them as meta-data for an ontology. Second, tools can produce different *views* of an ontology, according to the user's specifications. The view can take into account the user's expertise, his perspective, the required level of granularity, or a subset of the domain covered by the ontology that he is interested in. The information on which parts of the ontology are appropriate for expert or novice users can be part of the metadata provided either by authors or users. Tools can then use this information to present an appropriate view. Third, information on the ontology *reuse* by other ontologies can be collected automatically by crawlers. Swoogle, for example, ranks ontologies according to the number of references from other ontologies in its repository (Ding *et al.* 2004).

In addition to the information provided by authors and collected by automated tools, ontology users should be able to provide annotations and reviews. The information provided by ontology users is the focus of this paper. This information comes in different categories, and can come in three different forms: (1) a numeric ranking of a particular property of an ontology (such as correctness); (2) a set of keywords from an extensible controlled vocabulary (such as the domain of an ontology, or a list of tools used for its verification); and (3) free-text review of a particular dimension of an ontology.

The topics and dimensions on which we can collect information that we can collect from users of an ontology repository includes:

General review and rating: Users can provide text description of their experience with the ontology, a single general rating of an ontology, other comments that do not fall into any of the other, more specific, categories.

[4] http://protege.stanford.edu

Usage information: Which applications has the ontology been used in and was the use successful? Successful use of an ontology in a large number of application is a good indication of its quality.

Tool verification: Has the ontology representation (its syntax or semantics) been verified by any formal tools or methodologies? These tools can include anything from running a Description Logic reasoner to make sure that there are no inconsistencies in the ontology, to using methodologies such as OntoClean (Guarino & Welty 2002), to using a parser to make sure the syntax is correct (e.g., there is a large number of OWL ontologies available on the Web today that were created in a text editor and have syntax errors).

Coverage: Does the ontology cover properly the domain it claims to cover? Are there major gaps? Are some parts of the ontology developed better than others?

Correctness: Are there errors? What types of errors?

Concept-specific comments: Are there problems with specific concepts and concept definitions in the ontology? What alternative definition should be used?

Mappings to other ontologies and terminologies: How is the ontology related to other existing knowledge sources? The mappings could be as simple as a set of one-to-one correspondences between some concepts, or they can be complex declarative mappings (Crubézy & Musen 2003). Users can create these mappings manually or use assistance from a number of available semi-automatic tools to create them (Noy 2004).

Examples: One of the most useful kind of annotations are examples illustrating both what can and cannot be stated using the ontologies.

Citations and References: While many ontologies may be large, some of the most important ontologies and vocabularies tend to be very compact and the size of the ontology does not reflect the design effort that has gone into it. Annotations are a useful mechanism for linking ontology nodes to related publications.

Note that authors themselves can provide some of the information in the preceding list (such as mappings, usage examples, or verification status), just as users can fill in some of the details from the information that authors usually supply (such as ontology language or availability). In addition, the annotations and reviews can pertain to the ontology as a whole, or some of its parts. The level of granularity of annotation can be as small as a single concept or a set of concepts in an ontology.

Open Rating Systems

The user-supplied reviews and annotations of ontologies that we described in the previous section are inherently subjective. Not only users' expertise differs, but also different users prize different qualities in an ontology. For the information collected from other users to be useful, it has to be *personalized* for the user of the evaluation. At the same time, given how time-consuming the evaluation process can be, we would like to make the system for collecting these evaluations open. That is, we should be willing to collect evaluations from anyone who offers them. While such an open system is conducive to collecting more evaluations, it is inevitable that some significant portion of it will be of low quality. So, for the system as a whole to be usable, we have to both personalize it and *filter* it. In this section we introduce webs of trust as a mechanism for accomplishing both.

Open vs Closed Systems

We can broadly categorize publishing systems into two kinds—open and closed systems. In a closed system, there is a central authority who acts as a gatekeeper for publishing into the system. The gatekeeper typically verifies the quality of the content, and vouches for it. Such systems, by virtue of their central administration, tend to exhibit uniformity and predictability in the quality and growth rate of content. While some ontology repositories today are closed publishing systems (e.g., the Protégé OWL ontology library[5]), the inherent difficulty of evaluating the quality of an ontology and the subjectivity of such evaluation, leads to widely varying quality of ontologies in the repositories that work as a closed publishing system.

In contrast, the Web itself is an Open Publishing System. In an Open Publishing System, anyone can publish content into the system without going through a central gatekeeper. Lacking a central quality control mechanism, they also exhibit a wide variation in the quality of the content available. Such systems inevitably run into the problem that a large amount of the content available on them is of a low quality. This problem is encountered in every widely adopted open publishing system, ranging from the Usenet and bulletin boards to the World Wide Web. Open ontology repositories, such as DAML ontology library, Ontolingua ontology library, and others, certainly exhibit this varying quality of content as well.

As we have discussed, given the inherent difficulty and subjectivity of ontology evaluation, ontology repositories on the Semantic Web will inevitably work as an open publishing system. Consequently, filtering and ranking the content based on its quality and reliability becomes very important for these systems to remain usable. Therefore we will have to rely on users to rate the quality and usability of available ontologies. There

[5] `http://protege.stanford.edu/plugins/owl/ontologies`

are two main variations of collecting user reviews and annotations, based on whether the system for recording and publishing the ratings itself is open or closed.

Closed Rating Systems: In such systems, a group of "editors" are pre-qualified so that their ratings are known to be of an adequate quality. The Yahoo! and LookSmart directories are examples of such systems. In general, the biggest problem faced by such systems is that of scaling. If the amount of underlying content increases dramatically, as it did on the Web, such systems are unable to cover any significant portion of it. Not surprisingly, this problem is even worse for systems that rate ontologies and other complex cognitive artifacts: The task of evaluating ontologies is extremely difficult and time-consuming; whether or not an ontology is good depends highly on specific user requirements. As a result perhaps, researchers are not willing to be dedicated reviewers of ontologies that they do not plan to use.

Open Rating Systems: One solution to the problem of getting ratings on a corpus of rapidly growing content is to make the system for publishing these ratings itself open. Not only can anyone publish content, they can also publish ratings on this content. The systems for publishing the content and the ratings can be the same or different.

The scalability of the closed rating systems for ontologies and other similar artifacts is compounded by the fact that it is difficult to evaluate an ontology properly (as reviewers in a closed rating system can be expected to do) without actually using it in an application. However, in practice, researchers and developers need only a very small number of ontologies for their applications. Hence, each reviewer can be expected to provide quality reviews only for a small number of ontologies.

Over the last few years, a number of popular systems, such as Epinions and Slashdot,[6] which use the latter approach, have emerged on the Web. Google,[7] by virtue of PageRank (Page *et al.* 1998) interpreting links as ratings, also arguably shares this philosophy.

When anyone can publish ratings, two problems need to be solved:

Aggregation: We need a mechanism for aggregating the ratings of many sources into a single ranking. Various techniques such as PageRank (Page *et al.* 1998) and Rank Aggregation (Dwork *et al.* 2001) have been proposed for aggregating ratings and rankings.

Meta-Ranking: Given the open nature of the rating publishing system, often, there is a wide variation in the quality of the ratings themselves. Like the content being rated, many ratings turn out to be of poor quality. Consequently, some ranking and filtering needs to be performed on the ratings themselves. As a result, we are back to the problem of ranking, only this time, we are ranking ratings. Since each piece of content can have many ratings, this is a harder than the original problem of ranking the content.

[6] `http://www.slashdot.org`

[7] `http://www.google.com`

Systems such as Epinions, Amazon reviews and SlashDot have tackled the second problem by taking the open system philosophy to its natural next step. They allow ratings to be stated not just about the content, but also about the raters themselves. In practice, since a small, vocal minority of people state most of the ratings, and since there is considerable uniformity in the quality of ratings stated by a particular person, this step ameliorates the problem. These systems do have to avoid infinite regress: They must avoid getting into the question of whose ratings of raters we trust, and so on. Infinite regress is typically avoided by a combination of mechanisms:

- We can allow the hierarchy of rating statements to bottom out at some level. For example, we may disallow explicit ratings on ratings on raters.
- We can apriori designate some raters as trust-worthy and use these raters as the seed or root from which we compute the trust of all others. Advogato[8] and SlashDot use this approach.
- We can exploit the fact that many of the users of the system are themselves providers of ratings on raters: We can give each user a view of the overall system which is maximally consistent with their ratings. This approach also has the added benefit of giving each user a personalized view of the system.

Note that approaches that are based on *explicit* ratings and statements of trust that we describe here are in contrast to and complement approaches such as collaborative filtering. In collaborative filtering, the system treats the history of a user's behaviour as *implicit* ratings. As a result, and uses some implicit similarity between different users to infer more similarity.

Systems such as Epinions, SlashDot and Amazon reviews seem to work. That is, they support an Open Rating System, incorporating mechanisms for rating not just content but also other users, and using the combination of these two kinds of ratings they are able to do a good enough job of ranking that these sites have remained useful. We propose to use such a model for filtering and ranking evaluations of ontologies.

Such systems, which allow users to both rate any piece of content and each other, and which use these ratings and statements of trust to sort and filter results are referred to as *Web Of Trust (WOT)* systems. Technical details of WOT systems have been covered in other papers (Guha *et al.* 2004; Guha 2002). Here we summarize the main issues and approaches.

WOT Summary

Consider a WOT system for rating ontologies and their components. In such a system, we are given the follow-

[8] `http://www.advogato.com`

ing components:

- a set of objects (ontologies or other knowledge resources),
- a set of users,
- a set of reviews and annotations about the objects (each by a user),
- a set of ratings of the reviews and annotations, and
- a set of trust relations between users.

Given a user and his trust relations and a set of objects or a set of reviews, we have to identify the N objects (ontologies) or reviews that the user is most likely to review or rate positively. Many different approaches have been suggested for computing the top N objects.

TrustRank

Guha (2002) and others compute a $TrustRank$, analogous to PageRank (Page *et al.* 1998). A global $TrustRank$ can be used when the user does not have a WOT and a personalized version of $TrustRank$ (Haveliwala 2002) can be used when the user has a WOT. The primary advantage of this approach is that it works well even when the user does not yet have a WOT. The downside of this approach is that computing a different $TrustRank$ for each user can be very expensive.

Linear Algebraic Formulation

Guha and colleagues (2004) formalizes trust computations in terms of a set of primitive trust propagation transforms that apply on a trust matrix. Given the trust matrix for a set of users, these transforms allow us to propagate trust. The authors compare various combinations of primitives and show how some work better than others. The propagation of the trust can be a time consuming operation, but can be done offline so that the fully propagated WOT is available when the top N items need to be selected.

Probabilistic Formulation

The probabilistic formulation attempts to provide a semantics for trust propagation. It assigns the following intuitive meaning to the statement that Jane trusts Fred. If Jane trusts Fred, all other things being equal, she assigns a much higher likelihood to a statement being true if Fred believes it than if he didn't. Note that she might not be convinced (i.e., assign it a probability of 1) that the statement is true. She simply assigns it a much higher likelihood. Or more generally, all other things being equal, the likelihood she assigns a statement strongly correlates with that assigned by Fred. We formalize this intuition as:

$$trust(A, B) \Longrightarrow (P_A(s) \ll P_A(s|believes(B, s))$$

where $trust(A, B)$ means that A trusts B, $P_A(s)$ refers to the apriori probability assigned by the agent A to the statement s, $believes(B, s)$ means that the agent B believes that s is true and $P_A(s|believes(B, s))$ refers to the probability assigned by A to s, conditioned on B stating that s is true. This says that B believing s causes a substantial change in the likelihood assigned by A to s. It does not say by how much.

To operationalize this definition of trust, we need to compute a set of conditional probabilities for each user, corresponding to each possible set of trust relations between the remaining users. If we can assume independence between the trust relations, it can be shown that the linear algebraic formulation is an approximation to the probabilistic formulation.

Breadth First Search

Most large scale implemented systems (such as Epinions) use a much simpler approach to compute the top N objects. Given a user with a sufficient WOT, we do a breadth first search in the WOT around the user, collecting objects or reviews that have been rated highly, until we have N objects or have searched upto some cut off depth. For the case where the user does not have a WOT, these systems use a set of area experts who are assumed to be trust worthy.

Protégé Web Browser: Ontology Browsing and Annotation Tool

As we have discussed earlier, we envision that there will be ontology repositories on the Web that will collect ontologies and their metadata and maintain information about users, their reviews and annotations, and their webs of trust. A visitor to such repository will be able to search for ontologies satisfying his criteria and using his Web of Trust. The user should then be able to browse the top ontologies returned by the search and provide additional annotations and reviews. These annotations can describe features of an ontology as a whole or some components of it.

We will now discuss a working prototype of a component of such system that allows users to browse ontologies in their Web browsers and provide annotations for concepts in the ontologies. This browser is based on the Protégé ontology-management and knowledge-acquisition system developed at Stanford Medical Informatics.[9] From its inception, one of its primary goals was to provide a knowledge-base–editing environment that domain experts, and not just knowledge engineers, can use. It is arguably one of the most popular ontology editors today, with more than 20,000 registered users. While Protégé itself is a stand-alone Java application, one of its extensions is a version that runs in a Web browser. The Protégé Web Browser[10] allows users to use a standard Web browser to browse a Protégé ontology loaded on the server. It uses many of the same

[9] `http://protege.stanford.edu`

[10] `http://protege.stanford.edu/plugins/protege\_browser/index.html`

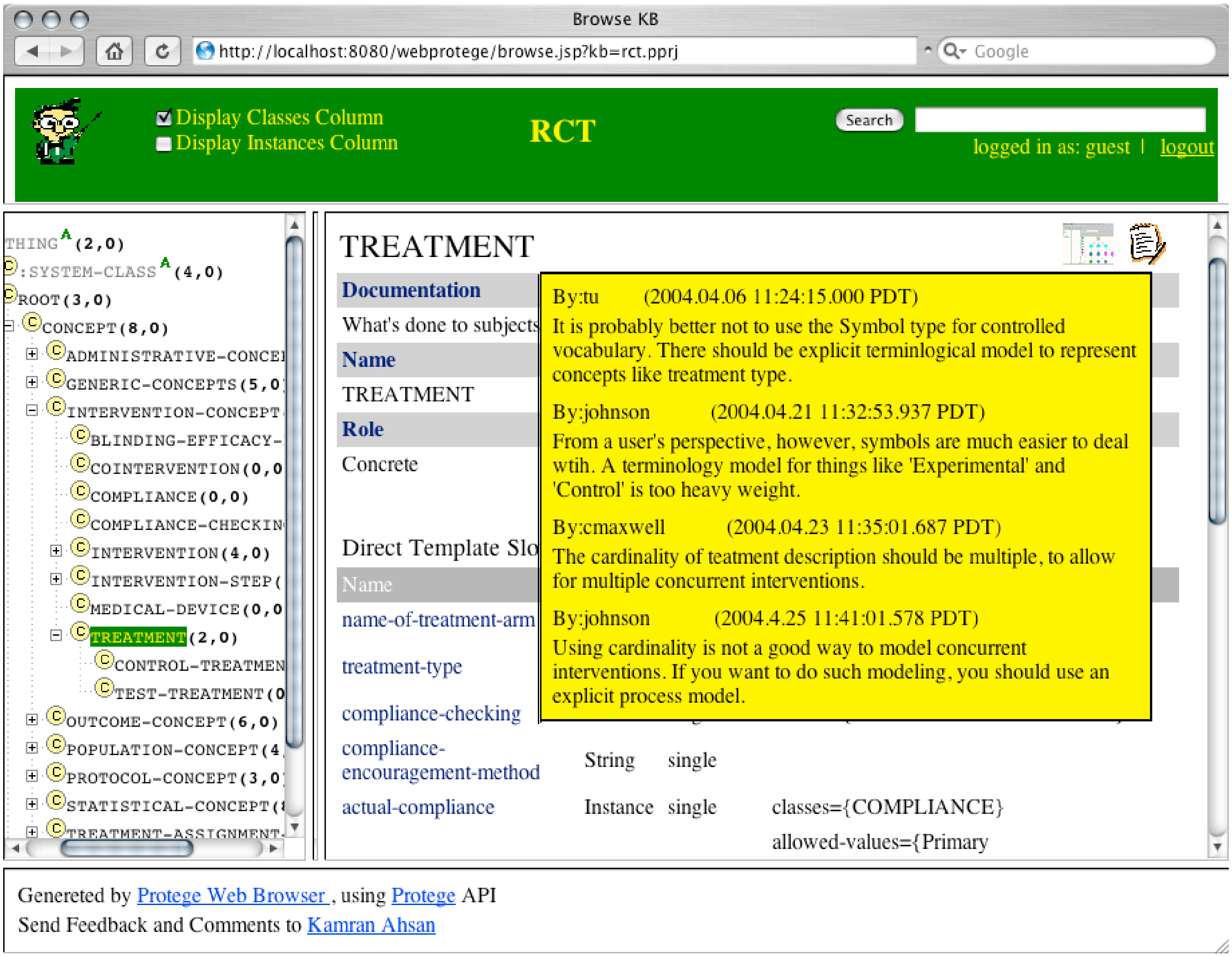

Figure 1: The Protégé Web Browser: The browser-based interface allows users to navigate the class hierarchy of the Protégé ontology (left column) and to view class definitions (right column). Moreover, users can create annotations for concepts in the ontology and view annotations made by others.

user-interface metaphors that the main tool uses. In addition, it allows users to annotate each frame in the ontology with free-text comments (Figure 1). The comments are then visible to everyone browsing the same ontology. If a new user decides to annotate the same frame, his comments are added to the list. This system is a prototype of an open-rating system where everyone can annotate any knowledge element in the ontology with a comment.[11]

While such system is sustainable and useful within a small closed project, it does not scale up when opened up to the Web. For example, imagine a combination of swoogle with the browser based version of Protégé in which any user on the Web can leave any kind of annotation on any ontology. While such an open system is almost a pre-requisite for collecting an adequate number of reviews and annotations, the noisy and spam-infested nature of the Web will quickly lead to a situation where a large number of reviews and annotations are not useful. The WOT-based approach of automatically identifying the reviews and annotations that are most likely to be relevant can be very useful in such a system.

Open Issues and Future Directions

We have discussed an Open Rating System for providing user reviews and annotations of ontologies available in repositories on the Web. We have outlined the main ideas underlining such system. There are many inter-

[11]The server is implemented in such a way that it can be configured to restrict the set of users who are allowed to make comment, thus making it a closed-rating system.

esting open issues and future research directions in this work.

Getting the critical mass Perhaps the most critical open issue is whether or not we can get the critical mass of people to review and annotate ontologies to make such system workable. At the time of this writing, the word "ontology" appears on almost three million Web pages; the Swoogle crawler indexes more than 300,000 ontologies and knowledge bases on the Web. Protégé has more than 20,000 registered users. These numbers may be an indirect indication of the number of people who may be interested in ontologies. One issue for future work is determining what exactly is the critical mass—how many people do we need—to have a useful system.

Issue-specific trust. We have discussed using a single *trust* value between two users A and B. In the context of ontology evaluation, one can imagine having different webs of trust for different topics and axes along which the users rank ontologies. For instance, A may trust B in his coverage ratings for ontologies dealing with biology and medicine, but value his opinion much less when it comes to ontologies in the domain of finance. Similarly, C may trust D on issues relating to the formality of an ontology, but perhaps not its practical usefulness. More specifically, the axes could include coverage for specific domain, conformance to and proper use of ontology languages, degree of formality, use in application, and so on. We can expand the model for computing the TrustRank to include several trust values for each pair of users, depending on the specific axis.

Corpus-based techniques. A system that we outlined will contain a large corpus of information that machine-learning tools can exploit. For instance, one type of information that we propose to collect are the mappings between concepts in different ontologies. Researchers have studied using corpus of matches between ontologies and schemas to discover new matches (Etzioni *et al.* 2003; Madhavan *et al.* 2005) or composing existing matches to create new ones (Kementsietsidis, Arenas, & Miller 2003; Madhavan & Halevy 2003; Aberer, Cudré-Mauroux, & Hauswirth 2003).

Annotation of other knowledge elements. We discussed mostly annotation of ontologies and their components. One can easily imagine using this idea for other elements in knowledge and data-intensive applications. Users can evaluate quality of specific data sets, repositories themselves, or even research papers. Open-rating systems have been very successful in the general domains on the Web. We believe that it can be equally successful in annotating knowledge-based products and other research artifacts. In fact, given the inherent subjectivity in such evaluations, it may be even more successful in this domain.

Acknowledgments

Kamran Ahsan has developed the Protégé Web Browser. We are grateful to Kaustubh Supekar for his thoughtful comments. This work was supported in part by a contract from the US National Cancer Institute. This work was conducted using the Protégé resource, which is supported by grant LM007885 from the United States National Library of Medicine.

References

Aberer, K.; Cudré-Mauroux, P.; and Hauswirth, M. 2003. Start making sense: The chatty web approach for global semantic agreements. *Journal of Web Semantics* 1(1).

Arpírez, J.; Gómez-Pérez, A.; Lozano, A.; and Pinto, S. 2000. Reference ontology and $(\mathrm{ONTO})^2$Agent: the ontology yellow pages. *Knowledge And Information Systems* 2(4):387–412.

Crubézy, M., and Musen, M. A. 2003. Ontologies in support of problem solving. In Staab, S., and Studer, R., eds., *Handbook on Ontologies*. Sringer. 321–342.

Ding, L.; Finin, T.; Joshi, A.; Pan, R.; Cost, R. S.; Sachs, J.; Doshi, V.; Reddivari, P.; and Peng, Y. 2004. Swoogle: A search and metadata engine for the semantic web. In *Thirteenth ACM Conference on Information and Knowledge Management (CIKM'04)*.

Dwork, C.; Kumar, S. R.; Naor, M.; and Sivakumar, D. 2001. Rank aggregation methods for the web. In *World Wide Web*, 613–622.

Etzioni, O.; Halevy, A.; Doan, A.; Ives, Z.; Madhavan, J.; McDowell, L.; and Tatarinov, I. 2003. Crossing the structure chasm. In *Conf. on Innovative Database Research*.

Guarino, N., and Welty, C. 2002. Evaluating ontological decisions with OntoClean. *Communications of the ACM* 45(2):61–65.

Guha, R.; Kumar, R.; Raghavan, P.; and Tomkins, A. 2004. Propagation of trust and distrust. In *International World Wide Web Conference, 2004*.

Guha, R. 2002. Open rating systems. `citeseer.ist.psu.edu/694373.html`.

Haveliwala, T. H. 2002. Topic-sensitive pagerank. In *WWW*, 517–526.

Kementsietsidis, A.; Arenas, M.; and Miller, R. J. 2003. Mapping data in peer-to-peer systems: Semantics and algorithmic issues. In *ACM SIGMOD International Conference on Management of Data*, 325–336.

Madhavan, J., and Halevy, A. 2003. Composing mappings among data sources. In *International Conference on Very Large Databases (VLDB) 2003*.

Madhavan, J.; Bernstein, P.; Doan, A.; and Halevy, A. 2005. Corpus-based schema matching. In *Proc. of the 18th IEEE Int. Conf. on Data Engineering (ICDE)*.

Noy, N. 2004. Semantic integration: A survey of ontology-based approaches. *SIGMOD Record* to appear.

Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1998. The pagerank citation ranking: Bringing order to the web. Technical report, Computer Science Department, Stanford University.