Phd Thesis: Provenance and Trust Abstract

Daniel Garijo, Oscar Corcho, Asunción Gómez-Perez
Ontology Engineering Group. Departamento de Inteligencia Artificial.
Facultad de Informática, Universidad Politécnica de Madrid.
Campus de Montegancedo s/n.
28660 Boadilla del Monte. Madrid. Spain
dgarijo@delicias.dia.fi.upm.es, {ocorcho, asun}@fi.upm.es

1 Problem statement

The interest in data provenance and trust has been increasing in the last years and the community is putting now a lot of effort in finding a standard model representation. The W3C provenance incubator group is focused on this area, analyzing different provenance models and making mappings between them and the Open Provenance Model (OPM)[1], which is the model they intend to make the standard.

We want to develop a provenance system based in OPM and a trust algorithm from that provenance information. Our aim will be a platform that will not store the contents generated by the users, but it will store all the references to them, the opinions from the users, information from social networks, etc. to obtain semantic information from the Web. In this context, being able to predict the trust of a source or being able to track the content we've generated is a great choice for any user.

2 Main questions of the thesis

The main research questions are the development of the provenance model reusing OPM and the development of a trust algorithm from provenance information. We have decided to choose the OPM as top level ontology because it is being set as a standard by the W3C, and many popular provenance vocabularies (such as Provenance Vocabulary[2], Provenir Ontology[3] or Dublin Core) are being mapped with it in the W3C provenance incubator group. Also, using it will make part of our model aligned with the other vocabularies, and therefore compatible with the data annotated by them.

On the other side, it is true that some of the mentioned vocabularies already address some of the problems we intend to solve (like the tracking of the data created in the web[2]), but none of them cover our complete domain. Furthermore, there is no trust algorithm based on the OPM provenance metadata yet.

The ideas proposed in this document are innovative because the provenance model will be one of the first vocabularies to use as reference the OPM, and the trust algorithm will be the first one based in the same top level ontology. This is an advantage, because the OPM is being set as a standard, and since mappings with the

most used vocabularies are being done in the W3C incubator group, our model would be automatically aligned with them.

3 General approach

To achieve our objectives, we have divided the approach in two parts: the development of a provenance model and the trust algorithm.

3.1 The provenance model

For the first part the work will focus in the development of an ontology reusing OPM and extending it to adapt it to our problem. We will have to model two main areas: the provenance of users and the provenance of objects created or used by those users. The former refers to the interaction between the users: their comments posted to other users, their opinions about other users, belonging to a group, etc. On the other hand, the latter refers to the history of every single piece of information in the platform: how it changed, who changed it, when, which sources where used to change it, etc.

Since there are many vocabularies modeling provenance in different contexts, it is important to know how to reuse the existent vocabularies for our purposes. Vocabularies such as Dublin Core, SWP, PML, WOT or Provenance Vocabulary are currently being mapped to OPM in the W3C Provenance Incubator Group.

In addition, some of these vocabularies cover part of the domain we want to cover: the Provenance Vocabulary [2] is designed to be used in the context of the web of data, (which is similar to part of ours), but it doesn't capture the relations between the users, their ratings, comments or groups. We will have to analyze the necessary concepts we will need to capture to determine the trust value, and if we can reuse some parts of the previous vocabularies to speed up the process (the mappings to OPM aren't always exact match of concepts between vocabularies). The advantage of having other vocabularies mapped to OPM is that the data annotated by them can also be reused by our vocabulary.

In a first phase our model will be designed to work in a closed environment, recording all the provenance metadata created and modified in the platform, but not outside of it. The model has to be robust and easy to expand and modify, being able to add or remove mappings from different vocabularies. In a second phase, we will also try to integrate provenance data from sources coming from outside the platform, like sources from Linked Data.

Once we have determined how our ontology model will be, we will develop a query system and a store system to be able to save and retrieve the provenance data.

3.2 Trust algorithm

The next step is to build a trust algorithm based on the stored provenance metadata. To do so, first we will have to determine the parameters that we will have to evaluate to assess the trust of a piece of information.

Then, we will study the relations between the selected parameters and their weights to calculate the trust value. The trust value of a piece of information is a number with range [0..1], where 0 means that the information we are dealing with is untrusted and 1 that it is trusted. We will also have to deal with incompleteness, estimating some of the parameters if the provenance graph doesn't provide enough information.

With these weighed parameters, now we will be able to measure the trust value of a provenance artifact, but we will still have to analyze how to update the provenance graph with the calculated trust value, either propagating the trust value through the graph or calculating the value each time.

The trust value is not a weighed sum of the parameters taken into account. Each of those parameters is calculated from the provenance graph, but in a different way. Some need preprocessing of the information of the object (like specifity), some may come directly from the annotations of the creator (like the date), and to determine some others (like bias) we may need to go deeper in the provenance graph or retrieve other information uploaded by the source, such as if it has submitted biased information before. The weight is the way we have to make some parameters more relevant than others.

This approach focuses on trust on information and its provenance, instead on trust on users like [6], which calculates the trust of a resource from the trust of the creator. It could also be nice to study how to combine our approach with the previous mentioned, for example associating the trust of the creator to the pieces of information it has created and taking it to account as another relevant parameter. As happened with the provenance system, the trust algorithm is required to be fast and scalable.

4 Proposed solution

This section is also divided in two parts. The first one introduces the provenance model, while the second one focuses on the trust algorithm.

4.1 The provenance model

The provenance model will be an ontology, using OPM top level ontology as reference. As we described in section three, it will be divided in two main modules: the first one focused on the provenance related to users and their interaction, and the second one capturing the provenance of data.

4.2 Trust algorithm

There exist many approaches of how to calculate trust in the Web, but most of them are focused in trust between agents or trust of the sources that produced a certain piece of information. There are only a few approaches of trust from the provenance of the information (like [4]), which is our main objective.

There also exist a wide range of possible trust parameters (identified in [5]), which can be organized in four different groups: authority, associated sources, provenance and bias. In the context we are dealing with, the possible trust parameters to take into account are the next:

Authority: If the entity that creates the information is an authority, we might automatically trust the information. It is highly related with the topic of the information and the bias.

Topic: If the piece of information we are evaluating can be used or not for the trust. In our context, users can rate and post comments about the different pieces of information. It would be great to use those comments to build up the trust.

Popularity/recommendation: The ratings and votes from the users are a key aspect to determine the final trust value.

Bias: Despite it is very hard to determine, bias is one of the most important aspects to take into account when calculating trust. We call bias the interest of the source or external agents to make us believe that certain information is true or fake.

Specifity: Normally when we deal with specific content we trust it more. The only way to determine this parameter is to preprocess some of the content of the information we're dealing with.

Age and recency: Recent information is more trusted than old information. To calculate this parameter we will have to look when was the information published, the expiry time of the data or the volatility of the data as in [4].

We will have to study the relations between these parameters to determine how they depend on each other and how can we assign them values in order to calculate trust. The trust algorithm will use the values extracted from the provenance information, so the parameters used will have to be part of the model. As a baseline, we will adapt to our provenance model the algorithm proposed in [4], based on the "timeliness" of an object; and then we will extend it with the proper estimations of the rest of the parameters.

We will also compare this proposal to other possible approaches, such as building a classifier with automatic learning techniques (like a Bayesian network), or the combination of fuzzy logic with the results obtained for each of the parameters (for example, if trust is a value in range 0..1 and we decide that a trusted information has to score at least 0.7, if it scores 0.699 can we consider it untrusted?).

It is also remarkable that trust is a subjective measure. It would be great if the user could modify the thresholds or weights of the parameters according to what she or he considers more relevant calculating trust.

After we decide how these parameters determine the provenance of an artifact, we can either propagate the trust value of the object through the graph, or we can choose not to do it, saving the trust value just for the current node. The first decision would lead to an updated graph, but on the other side it wouldn't be very efficient (every time we calculate the trust of a piece of information, it would have to propagate it again through the graph). The second approach would be more efficient, but we would have to calculate the provenance of an object every time. We must not forget that a trust value doesn't last forever, so every trust value must be produced with an expiry date, or an expiry condition. If the condition is broken or the expiry date is over, the trust value will be worthless and it will have to be calculated again.

5 Evaluation

First of all we will have to evaluate the provenance model. To do so, we will use the system to answer the queries proposed in the First Provenance Challenge. These queries were made over a simple provenance graph, and they are supposed to cover most of the general and specific queries a user can make over a provenance graph.

For the second part, we will have to check that the trust values calculated by the system are consistent with the parameters with more weight. We will start with a small provenance graph covering various examples, and we will confirm manually that the calculated values attach more importance to the relevant trust parameters. We will also have to evaluate the scalability of the system: since it is based on the previous module, the scalability here will focus in the time the algorithm uses to calculate the trust once it has retrieved all the needed information from the graph.

As baseline, we will use the database used in [4]: the FlyTed Database, to test if our adaptation of the trust algorithm to our provenance model is correct and to compare the results obtained. When available, we will test the complete trust algorithm with the data produced in our project.

6 Future work

We are in the first phase of development of the thesis, so the future work is to complete the provenance prototype and keep up with the investigation of how to calculate the value of the trust parameters. Once done, the next steps are to complete the trust algorithm and evaluate its performance. As we have said before, it would also be nice to integrate metadata from Linked Data, annotated with other vocabularies through mappings to OPM, and be able to calculate trust parameters from it too.

References

1. Luc Moreau, Ben Cli_ordb, Juliana Freirei, Yolanda Gil, Paul Groth, Joe Futrelle, Natalia Kwasnikowska, Simon Miles, Paolo Missier, Jim Myers, Yogesh Simmhan, Eric Stephan, Jan Van den Bussche. "The Open Provenance Model" (v1.1). Future Generation Computer Systems. 2009.
2. Olaf Hartig. "Provenance Information in the Web of Data". WWW 2009.
3. Satya S. Sahoo, Roger Barga, Amit Sheth, Krishnaprasad Thirunarayan, Pascal Hitzler. "PrOM: A Semantic Web Framework for Provenance Mangement in Science". WWW 2010.
4. Olaf Hartig, Jun Zhao. "Using Web Data Provenance for Quality Assessment".ISWC 2009.
5. Yolanda Gil, Donovan Artz." Towards Content Trust of Web Resources". Journal of Web Semantics. 2007.
6. Jennifer Golbeck, Aaron Mannes. 2006."Using Trust and Provenance for Content Filtering on the Semantic Web". Proceedings of the Workshop on Models of Trust on the Web, WWW 2006.