Ontology Reuse Decision Support: Visualize the Ontology or its Usage?

Marek Dudáš and Vojtěch Svátek

Department of Information and Knowledge Engineering,
University of Economics, W. Churchill Sq.4, 130 67 Prague 3, Czech Republic,
{marek.dudas|svatek}@vse.cz

Abstract. When considering an existing ontology for reuse to annotate new data, one needs to check if the ontology contains classes and properties suitable for describing the data. Visualizing the ontology itself in some of the available tools, to support such a decision, is an obvious option. We propose that ontology usage visualization, a slight variant of dataset schema visualization, might however be an interesting, complementary, alternative. We test the idea in a small evaluation with users.

1 Introduction

When seeking a suitable ontology for semantically describing a prospective RDF dataset, one can often identify several candidates at portals such as Linked Open Vocabularies (LOV) [8]. However, we then need to find out, for each candidate ontology, whether it is indeed suitable in the sense of containing classes and properties necessary to represent our data; this task can be characterized as *model coverage* assessment. An obvious option is to leverage on ontology visualization tools. In this paper we examine the possibility of employing *ontology usage visualization* (OUV), as an alteration of the recently emerged dataset (or endpoint) schema visualization,[1] for the same task, and present a small empirical study comparing these two approaches.

We performed an experiment with a group of users, whom we asked to check the model coverage of selected ontologies. They used LODSight [3], an OUV tool developed by us, and WebVOWL [6], a state-of-the-art ontology visualization tool. When using WebVOWL, the users were also allowed to look into the ontology documentation if available – to simulate what we consider a common approach to model coverage checking.

2 Ontology Visualization and Ontology Usage Visualization

Ontology visualization Ontology visualization tools, as surveyed, e.g., by Katifori et al. [5], take the OWL code of the ontology as input. They usually display it as a node-link visualization where nodes represent classes and links represent properties – based on domain/range definitions.

[1] Different terms like 'visualization of schema information from Linked Data' can be found, as the terminology is not yet stable. As regards 'ontology usage visualization', it is actually a new term only coined by us, for the moment; we are not aware of any explicit term already in use.

WebVOWL WebVOWL is a web-based OWL ontology visualization tool. It uses a node-link visualization technique with force-directed layout. The nodes, differentiated by color and shape, represent the OWL classes (and instances), and the links show the relationships between them, of which domain/range and subClassOf are most typical. An example of a partial visualization of FOAF ontology[2] is shown in Figure 1. The main reason for choosing this tool is that URLs leading to WebVOWL visualizations of selected ontologies can be constructed easily.

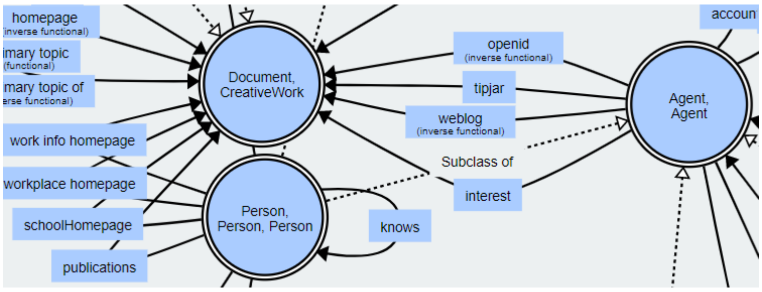

Fig. 1. Partial screenshot of FOAF visualization in WebVOWL.

Ontology usage visualization The goal of ontology usage visualization (OUV) is to show how entities from an ontology are used in existing datasets. It can be used (1) to find out whether an ontology is suitable for modeling the given problem (i.e., checking model coverage, the use-case presented in this paper), (2) for learning how to use an ontology to annotate data, or (3) for detecting errors in the usage. Its input is one or more RDF datasets and the output shows all combinations of classes and properties that are used in the datasets. More precisely, classes whose instances are linked with the properties in the dataset are shown linked with the properties in the ontology usage node-link visualization. The result is similar to ontology visualization, yet it does not depend on domain/range relationships but on actual data in the dataset. The principle is the same as in dataset schema visualization provided by tools like LD-VOWL [9]. The difference is in the purpose. Dataset schema visualization is concerned with a single dataset and its goal is to help users find out what the dataset contains and how to query it. In terms of implementation, ontology usage visualization is a dataset schema visualization supporting the merger of several schemas into one visualization, filtering the visualization in order to focus on the selected ontology, and displaying all classes and properties found in the dataset, i.e., not only the most frequent ones, because we need to see all 'capabilities' of the ontology and not just the common usage.

LODSight LODSight[3] is a schema extraction and visualization tool for dataset creators and ontology engineers. It uses SPARQL to find type-property and datatype-property

[2] `http://xmlns.com/foaf/spec/`

[3] Version used for this paper is at `http://rknown.vserver.cz/lodsight`.

paths in one or more datasets. A (1-step) type-property path is a sequence `type1 - property - type2`, where the types are classes of instances connected by the property. Datatype-property paths are sequences of `type - datatype property - datatype`. All paths are merged into one graph and visualized in one view. All the OUV features are implemented: subsequent merging of graphs coming from different datasets is possible. Filtering the visualized graph to nodes coming from a selected ontology and nodes directly linked to them is also implemented, as well as the "maximum detail" option leading to visualization of all classes and properties found in the graph. An example of FOAF usage visualization is shown in Figure 2.

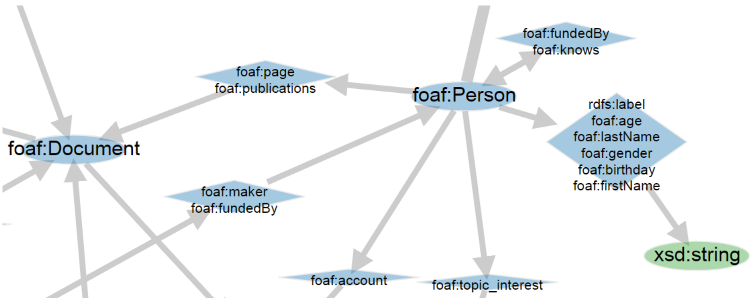

Fig. 2. Partial screenshot of FOAF ontology usage visualization in LODSight

Model coverage When an ontology contains suitable classes and properties that can be used to describe the relationships and entities in the given data model[4] we say it *covers the model*.[5] A class or property is 'suitable' when it has the same or more general, but not too distant, meaning as the concept in the given situation. The 'not too distant' is subject to human consideration. For example, a class Author can obviously be used as a type of an instance representing an author of a document, as well as more general classes Person or Human. A class such as Entity would be too general and thus unsuitable.

3 Related Research

We are not aware of any research dealing with model coverage analysis directly. Somewhat related is the ontology coverage check designed by Pammer et al. [7]. However, its purpose is different and the principle reversed. It uses test data (instances) to find out whether it covers all concepts in the ontology in order to find errors in the ontology.

Several dataset schema visualization tools similar to LODSight exist. However, none of those we are aware of explicitly supports ontology usage visualization. The most recently presented tools are LD-VOWL and that by Florenzano et al. [4]. Both

[4] The model can be in any form, e.g., implicit in the engineer's mind, a free text description, or a data sample (typically as a table).

[5] This notion of 'covering' is similar to that of Chalupsky [2].

use a set of SPARQL queries to summarize a dataset in a similar way as LODSight. LD-VOWL works in real-time thanks to limiting the visualization to the most frequent classes and properties. The latter tool uses precomputed summarization (as LODSight does) but in contrast to LODSight allows incremental exploration of the visualization. Both systems take into account subClassOf relationships, which LODSight does not.

4 Experiment

The hypothesis we tested was that using LODSight for model coverage analysis of ontologies will not require more time and won't be more erroneous than using WebVOWL, where errors are wrong assumptions whether an ontology covers the given model. Due to low number of participants, we did not expect to statistically confirm the results. We therefore focused also on possible causes of specific errors made by users in both tools.

4.1 Setup

Users and experimental protocol The test users were 13 students at the end of an ontology engineering course; all thus had basic knowledge of OWL and tools like Protégé. They were given a 30-minute explanation of the idea of model coverage assessment, followed by two 15-minute hands-on tutorials of WebVOWL and LODSight. The experiment directly followed the tutorial. The students were given two consecutive tasks. First, they had to analyse 6 ontologies in order to find out whether they can be used to say that "someone is an author of some article". 6 (randomly chosen) students were asked to use WebVOWL and textual ontology documentation, while the remaining 7 used LODSight. In the second task, the students were given 4 different ontologies and were asked to select those allowing to express that "a conference is held at some location". The first group of 6 students now worked with LODSight and the remaining 7 students who had used LODSight in the first task now used WebVOWL.

The assignment documents[6] included ontology IRIs, links to their WebVOWL visualizations and to their documentation (if available) for the first group and ontology IRIs and links to their LODSight visualizations for the second group. Both documents included also screenshots of the WebVOWL or LODSight visualizations, focused on their relevant parts. I.e., the screenshots, added as backup for technical difficulties, were arranged to show classes and properties relevant to the assignment.

All students worked in the same room and started at the same time. The first author of the paper was present the whole time to answer technical questions and prevent the students from influencing each other. Students sitting next to each other were assigned different groups. When all students finished the first task, they were given a link to the document with the second assignment. After finishing each task, the users were also asked to fill-in a SUS questionnaire[1] about the tool they just had used.

Ontologies and data used The LOV dataset was used to find and select ontologies that at least partially cover the given model. The relationships in the assignments were

[6] Available from `http://protegeserver.cz/OUV-experiment`.

chosen based on simple heuristics that persons, authorship, conferences and locations are covered by many ontologies and therefore it should be easy to find enough data for the tests. This way, the following ontologies were gathered for the first task: `foaf`, `swportal`, `earth`, `npg`, `con` and `schema`.[7] All except the `con` ontology cover the assigned model. For the second task, the following ontologies were presented to the students: `bibo`, `swc`, `gnd` and `bibtex`. All of them except `bibtex` cover the assigned model.

A prerequisite of using LODSight is the existence of at least one dataset where the analyzed ontology is used. Ideally, multiple different datasets would be used for one OUV. We searched for such datasets using the datahub.io and stats.lod2.eu portals, with no success. Therefore we had to create sample datasets by hand. For each ontology we created example data representing the model given in the task plus some random data. We created only one dataset per ontology, since, in this manual scenario (only approximating what real usage of the ontologies could be) there was no reason for having more datasets per one visualization. The samples contained between 13 – 87 different properties and 6 – 55 classes. It was not feasible to use all classes and properties from each ontology in the sample datasets, which leads to a possible bias. Namely, the ontology visualization in WebVOWL shows all classes and properties from the ontology and is therefore more cluttered and harder to read than LODSight visualization just because of the number of displayed elements. However, we assume that it reflects a possible real-world situation because it is uncommon that a dataset with real data would use all classes and properties from an ontology either: it would simply use only those needed to represent the given data. The sample datasets were summarized in advance and the students were given direct links to LODSight visualizations of usage of each ontology.

4.2 Results

We measured the time needed to complete each task and the number of errors, i.e., of incorrectly classified ontologies. A brief summary is shown in Table 1. It shows the average, median, minimum and maximum values of time, the numbers of false positives (ontologies incorrectly classified as covering the given situation) and false negatives, precision, recall and the F-measure. The aggregate values were counted per each combination of task and tool. The average time was lower with LODSight in both tasks. The average number of errors was lower compared to WebVOWL when using LODSight for the ‘conference-location’ task, but higher for the ‘person-author’ task. It appears as if the group that used VOWL for the ‘person-author’ task outperformed the second group in OWL understanding, as these users made fewer errors than the other group regardless of the tool they used. A larger sample of users would be needed to enable valid statistical tests.[8]

Errors discussion Analyzing the individual answers in detail,[9] we can see that most users were unable to correctly classify the `bibo` ontology using WebVOWL. That is

[7] We list only the prefixes/abbreviations as used in LOV, for brevity.
[8] No statistically significant difference was found using either t-test or Wilcoxon test.
[9] Available online: `http://protegeserver.cz/OUV-experiment`.

Table 1. Aggregated results from the experiment: time (h:mm:ss), false positives (FP), false negatives (FN), precision, recall and F-measure (F1).

	Conference-location using VOWL						Conference-location using LODSight					
	time	FP	FN	Precision	Recall	F1	time	FP	FN	Precision	Recall	F1
avg	0:10:40	0.50	1.00	0.82	0.61	0.67	0:07:24	0.57	0.43	0.85	0.86	0.83
min	0:07:35	0.00	0.00	0.50	0.33	0.40	0:03:25	0.00	0.00	0.67	0.67	0.67
max	0:13:52	1.00	2.00	1.00	1.00	0.86	0:12:36	1.00	1.00	1.00	1.00	1.00
median	0:10:35	0.50	1.00	0.88	0.67	0.73	0:06:24	1.00	0.00	0.75	1.00	0.86

	Person-author using (VOWL)						Person-author using LODSight					
	time	FP	FN	Precision	Recall	F1	time	FP	FN	Precision	Recall	F1
avg	0:19:30	0.00	1.00	1.00	0.83	0.89	0:14:31	0.14	1.71	0.97	0.66	0.76
min	0:17:39	0.00	0.00	1.00	0.40	0.57	0:07:26	0.00	0.00	0.80	0.40	0.57
max	0:23:18	0.00	3.00	1.00	1.00	1.00	0:24:22	1.00	3.00	1.00	1.00	1.00
median	0:18:37	0.00	1.00	1.00	0.90	0.94	0:12:51	0.00	1.00	1.00	0.80	0.80

probably because the property ‘place’ for linking a conference to its location does not have domain and range specified, and WebVOWL thus displays it far from the class ‘Event’. Similarly, users had problems to analyze the `swc` ontology, probably because the property ‘has location’ is displayed between the classes ‘Organized Event’ and ‘SpatialThing’, quite far from the class ‘Conference’ which is a second-level subclass of ‘Organized Event’. The users made many mistakes in case of the `bibtex` ontology, regardless of the tool they used. It is probably because the ontology contains a class ‘Conference’ and a property ‘hasLocation’, but the class represents a paper in conference proceedings rather than an actual conference. It can be seen from the fact that properties like hasISBN’ are used with its instances. Regarding the user answers in case of the ‘person-author’ task, there is the same pattern with `schema` as with `bibo` – it also misses domains and ranges and WebVOWL therefore does not display the properties in a user-friendly way. An open question is why `swportal` has been incorrectly classified by 5 of 7 users with LODSight while all 6 users analyzed it correctly with WebVOWL. The ontology is tuned for representing a *list* of authors, which might make it harder to understand for a beginner, but this aspect seems comparably troublesome in both visualizations.

SUS questionnaire The average SUS score is almost the same for both tools: 65 for LODSight and 63 for WebVOWL. This is coherent with the status of both tools as of relatively mature academic prototypes. The result indicates that the error rates is likely due to inherent features of OUV vs. ontology visualization rather than due to different usability of the particular tools.

5 Discussion and Conclusion

We tested the idea that OUV might be suitable for checking the model coverage of ontologies in addition to common ontology visualization and textual documentation in an experiment with users.

The results suggest that using LODSight is comparable to the common approach. The users made a similar number of mistakes and also the average SUS score is almost the same for both LODSight and WebVOWL. The users were faster when using LODSight, but that can be due to the additional time spent looking at the ontology documentation as additional resource when using WebVOWL. There are some specific cases when LODSight (and OUV in general) seems to be more suitable, especially when the ontology lacks domain/range relationships. On the other hand, an essential prerequisite for OUV are existing datasets with correct usage of the ontology. As our attempt to find data for our experiment suggests, those are very scarce.

We propose that both approaches can be simply combined. I.e., dataset creators analyzing an existing ontology might look at it in an ontology visualization tool and, if there are available datasets using the ontology, check it with OUV as well – especially when domain/range relationships are missing. Although we did not bring convincing results, we hope this paper encourages more research on OUV in the future.

Acknowledgments

The research has been supported by the H2020 project no. 645833 (OpenBudgets.eu).

References

1. Brooke, J., et al.: SUS – A quick and dirty usability scale. Usability evaluation in industry 189(194), 4–7 (1996)
2. Chalupsky, H.: Ontomorph: A translation system for symbolic knowledge. In: KR2000: Principles of Knowledge Representation and Reasoning. pp. 471–482 (2000)
3. Dudáš, M., Svátek, V., Mynarz, J.: Dataset summary visualization with LODSight. In: The 12th Extented Semantic Web Conference (ESWC2015). `http://lod2-dev.vse.cz/lodsight/lodsight-eswc2015-demopaper.pdf`
4. Florenzano, F., Parra, D., Reutter, J.L., Venegas, F.: A visual aide for understanding endpoint data. In: VOILA! 2016. p. 102
5. Katifori, A., Halatsis, C., Lepouras, G., Vassilakis, C., Giannopoulou, E.: Ontology visualization methods – a survey. ACM Computing Surveys (CSUR) 39(4), 10 (2007)
6. Lohmann, S., Negru, S., Haag, F., Ertl, T.: VOWL 2: user-oriented visualization of ontologies. In: Knowledge Engineering and Knowledge Management, pp. 266–281. Springer (2014)
7. Pammer, V., Scheir, P., Lindstaedt, S.: Ontology coverage check: support for evaluation in ontology engineering. In: FOMI 2006. The 2nd workshop: Formal Ontologies Meet Industry
8. Vandenbussche, P.Y., Atemezing, G.A., Poveda-Villalón, M., Vatant, B.: Linked open vocabularies (LOV): a gateway to reusable semantic vocabularies on the web. Semantic Web 8(3), 437–452 (2017)
9. Weise, M., Lohmann, S., Haag, F.: LD-VOWL: Extracting and visualizing schema information for linked data. In: Visualization and Interaction for Ontologies and Linked Data (VOILA! 2016). p. 120 (2016)