Visualizing the Drift of Linked Open Data Using Self-Organizing Maps

Albert Meroño-Peñuela[1], Peter Wittek[2,3], and Sándor Darányi[3]

[1] Department of Computer Science, Vrije Universiteit Amsterdam, NL
albert.merono@vu.nl
[2] ICFO-The Institute of Photonic Sciences, ES
[3] Swedish School of Library and Information Science, University of Borås, SE

Abstract. The urge for evolving the Web into a globally shared dataspace has turned the Linked Open Data (LOD) cloud into a massive platform containing 100 billion machine-readable statements. Several factors hamper a historical study of the evolution of the LOD cloud, and hence forecast its future: its ever-growing scale, which makes a global analysis difficult; its Web-distributed nature, which challenges the analysis of its data; and the scarcity of regular and time-stamped archival dumps. Recently, a scalable implementation of self-organizing maps (SOM) has been developed to visualize the local topology of high-dimensional data. We use this methodology to address scalability issues, and the Dynamic Linked Data Observatory, a regular biweekly centralized sample of the LOD cloud as a time-stamped collection. We visualize the drift of Linked Datasets between 2012 and 2016, finding that datasets with high availability, high vocabulary reuse, and modeling with commonly used terms in the LOD cloud are better traceable across time.

Keywords: Linked Data, Semantic Drift, Visualization

1 Introduction

The original vision of the Semantic Web [2] is becoming a reality by the publishing paradigm of Linked Data, which so far consists of a global, Web-accessible graph of knowledge of 100 billion RDF statements known as the Linked Open Data (LOD) cloud [4]. In this Web dataspace, *dynamics* are the norm: new data are added, old are removed, and schemata evolve to accommodate new requirements and changes in the domain. Terms like *semantic drift* are used to refer to problems that arise as a consequence of these dynamics. One of the typical access methods for Linked Data is SPARQL, a protocol and query language. Datasets become inaccessible, for instance, due to query templates that no longer correspond to the schema [8]. The maintainability of ontologies is cumbersome – large change logs are costly and client applications cannot adapt to them. It is

[4] http://lod-cloud.net/

impossible to have an abstract perception of the history of the LOD cloud, and it is equally infeasible to forecast how it might evolve in the future.

Recent work addressing some areas of semantic drift shows that solutions to specific problems in this domain can provide great real-world benefits, such as automated maintenance of biomedical ontologies [10] and efficient update of caches in applications using LOD [3]. However, the global challenge of semantic drift is emphasized by three inherent factors of structured data on the Web: *(1)* its unprecedented and unconstrained *scale*; *(2)* its highly *distributed* nature; and *(3)* the scarcity of regular and systematic *timestamped dumps*. Several approaches study semantic drift over time in the context of LOD and the Semantic Web, although none of them addresses all of these factors. For example, Wang et al. [13] provide theoretical ground for heuristics under the assumption of existing *timestamped dumps*, but do not address problems (1) and (2). Similarly, the well established field of ontology evolution [12] has seen a rebirth in approaches that model it using machine learning [9,10]. The Dynamic Linked Data Observatory (DyLDO) [5] covers (2) and (3), but does not solve the problem of scale. Recently, a massively parallel implementation of self-organizing maps (SOM) has proven useful to model, reduce the dimensionality of, and visualize large scale evolving data in an efficient way [6]). In this paper, we study the requirements derived from problems (1), (2), and (3) for visualizing Linked Data that evolve over time; and we argue that the combination of DyLDO and Somoclu fulfills these requirements in a comprehensive and visually explicit manner. We find that datasets published as LOD in the period 2012-2016 with high availability, high reuse of vocabularies[5], and high adherence to commonly used terms are better traceable in time. More specifically, the contributions of the paper are as follows:

- We describe requirements for visualizing the temporal drift of datasets in LOD (Section 3);
- We argue the convenience and describe the foundations of SOM techniques to address these requirements (Sections 2 and 4);
- We use a scalable implementation of SOM over the DyLDO dataset, a large crawl of timestamped LOD in the period 2012-2016 (Section 4). Datasets that were highly available with online and de-referenceable URI resources, reused existing vocabularies in the LOD cloud, and chose common and highly used terms to describe their resources are better traceable over time in low-dimensional representations of high-dimensional input Linked Data.

2 Related Work

The problems of semantic change and drift concern various research fields. In the areas of Semantic Web and knowledge representation, ontology evolution [7] addresses "the timely adaptation of an ontology and consistent propagation of changes to dependent artifacts" [1]. Features of evolution have been studied [12] and used for prediction using machine learning [10]. Gonçalves et al. [4] use

[5]See `http://lov.okfn.org/dataset/lov/`

Description Logics to calculate differences between ontologies (so-called *semantic diffs*). Wang et al. [13] define the semantics of concept change and drift, and how to identify them. General surveys of semantic change in other fields, including language, have recently appeared [11].

Caching Linked Data is the primary goal of repositories like LODCache[6]. To the best of our knowledge, the Dynamic Linked Data Observatory, DyLDO [5], is the only available crawl of LOD with rigorous timestamps.

Dimensionality reduction is a standard technique in statistical data analysis and machine learning. In sparse spaces, such as the ones we obtain from LOD, the global topology of the space is often less important than the local regions where many data instances have overlapping nonzero elements: techniques that focus on preserving local topology are preferred. Self-organizing maps are an example of this type [6]. They were introduced to the study of drifts [14] which was enabled at scale by a massively parallel implementation of the methodology [15]. After training a map, each data instance will have a matching point called the best matching unit (BMU) on the map – the immediate surroundings of this point reflect the local topology of the original space. Intense colours on the map indicate high distances between the original data points.

3 Requirements for Visualizing LOD Drifts Over Time

Table 1 compiles a set of requirements for visualizing the temporal drift of datasets in the LOD cloud. For each requirement, we assign a name, a description, and whether it concerns the drift analysis algorithms or the analyzed data. As a basis for this, we consider the following characteristics of the LOD cloud:

- **High-scale**. The size of the LOD cloud is large, counting 100 billion statements in an unconstrained environment that allows for arbitrary growth;
- **Distributed**. LOD uses the Web as a publishing platform, meaning that Linked Data can be potentially found in any node of the Web;
- **Timestamped dumps**. In LOD, temporal drifts may occur anytime. Since a sound and complete version control that manages all changes in LOD is currently impracticable, regular timestamped snapshots of the LOD cloud (or a sample of it) are necessary to study drift.

4 Experiments

We describe an empirical application of SOM to different subsequent snapshots of the LOD cloud over time, using the Somoclu algorithm [15] and the DyLDO [5] dataset[7,8]. Somoclu and DyLDO fulfill the requirements of Table 1, and hence their combination poses a promising solution for visualizing LOD drifts. Somoclu fulfills the algorithm requirements in terms of *efficiency*, *visualization*, *dimensionality reduction*, *topology preservation* and *unsupervised learning*; while DyLDO does for the requirements of *centralization*, *sampling* and *provenance*.

[6] `https://datahub.io/dataset/openlink-lod-cache`

[7] `http://swse.deri.org/dyldo/`

[8] All experimental data and code are available online at `https://github.com/albertmeronyo/somoclu-dyldo`

Name	Description	Alg./Data
Efficiency	Solutions addressing semantic drift in LOD should be efficient in order to address its scale.	Alg.
Centralization	Linked Data is scattered over the Web, and analysis of drift is impracticable in Web-distributed data. Datasets to be analyzed need to be centralized.	Data
Visualization	The result of the analysis of drift in LOD should be an understandable, meaningful and coherent visualization to the human eye	Alg.
Sampling	The distributed architecture of the Web makes complete studies of drift over all LOD very difficult. Samples of different sizes are needed.	Data
Dimensionality reduction	Most features of Linked Data are usually in a high-dimensional space (e.g. total number of URIs in predicate position). Representing these in a lower-dimensional space is necessary for human-understandable visualizations	Alg.
Topology preservation	Any technique reducing the dimensionality of the input space must preserve its topology in order to faithfully represent the subtleties of drifts in LOD	Alg.
Unsupervised learning	The high-scale of LOD makes the manual annotation of instances by humans impracticable, and algorithms analyzing drifts in LOD should operate in an unsupervised manner.	Alg.
Provenance	The Linked Data subject to the study of drift should have accurate and structured provenance information, especially regarding its collection time in the form of timestamps	Data

Table 1: Requirements for visualizing LOD drifts, concerning algorithms and data.

4.1 Preparation

We select 79 DyLDO snapshots comprised in the period from 2012-05-13 until 2016-03-27 (dates are in format `YYYY-MM-DD`). Since a study of drift that considers RDF triples as units of analysis would be both cumbersome and superficial, we consider URIs that appear in *predicate* position only. This allows us to simplify the matrix construction process, at the time we focus the analysis on the usage of vocabulary terms.

First, we build an index of common unique predicates (275,412) and common unique graph names (4,506) in all these snapshots. All these predicates and graph names occur in all snapshots. Secondly, for each snapshot we build a sparse matrix that indicates the frequency of each predicate in each graph name. These highly sparse vectors characterize the contents of the graphs: every instance is a vector containing counts of each possible predicate in that graph name. This means we generate 79 sparse matrices with 275,412 dimensions and 4,506 instances, with an average sparsity (i.e. non-zero elements) of 0.03%.

4.2 Training SOMs

Training self-organizing maps is a computationally expensive task. We rely on a massively parallel implementation called Somoclu [15][9]. This implementation

[9]See `http://peterwittek.github.io/somoclu/`. We use its Python interface for visualization, which is available at `https://pypi.python.org/pypi/somoclu/`

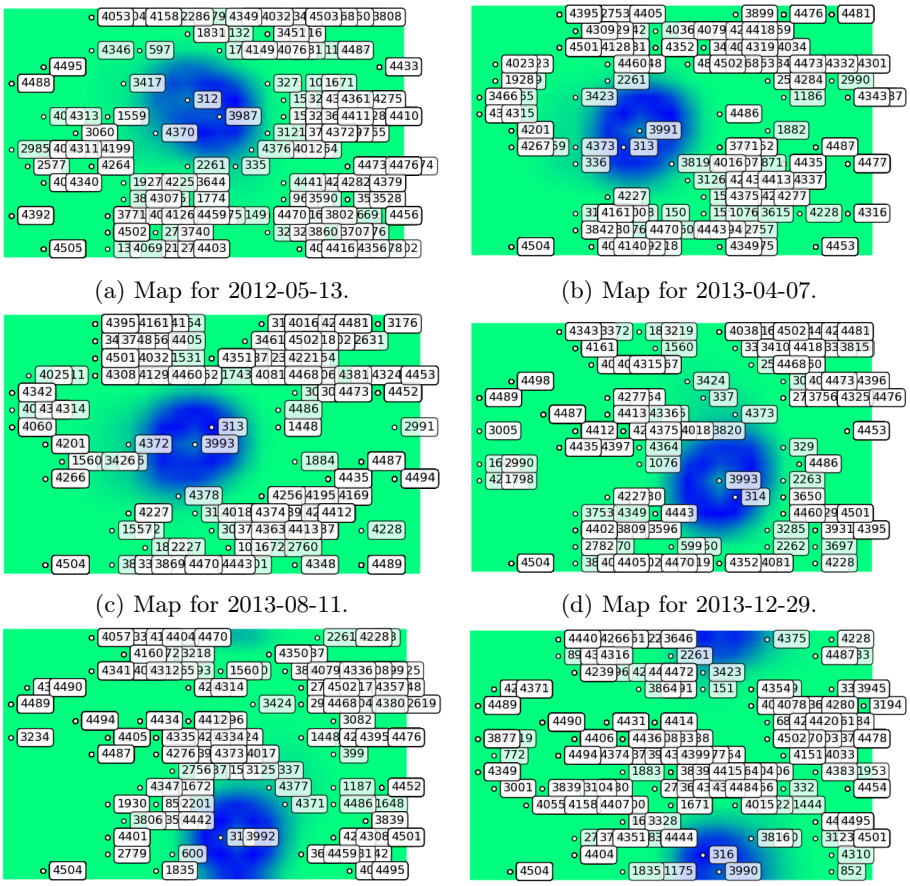
can also handle sparse input data. Apart from the computational requirements, a key challenge in using SOMs is the large number of parameters that require fine-tuning. As a rule of thumb, toroid topology yields higher quality maps because we avoid jamming best matching neurons along the edges as we would with a planar layout. We seldom see fundamental topological differences between hexagonal and square-shaped neurons, so we use the second for their easier visual treatment. Due to memory limitations, we establish a map size of 15 rows and 25 columns. To ensure a smooth initial map, we train it for a longer time (ten epochs instead of three) and at a higher learning rate (0.1 instead of 0.05) than the rest. We use the output neuron weights of snapshot t as a codebook to initialize the map of snapshot $t + 1$.

4.3 Results

Figure 1 shows a summary of 6 key frames in the evolution of the vocabulary terms in the DyLDO dataset from 2012-05-13 until 2016-03-27. The numbered labels correspond to the named graph's BMUs. Complete animations are available online[10]. The output maps of Figure 1 display the following chronology:

- 2012-05-13: There are at least three recognizable clusters in the middle of the map. Some BMUs arise in the centers of these clusters, while others appear in their neighborhoods. The named graphs that correspond to these BMUs are, by its proximity to the cluster's center:
  - 3987: `http://www.bbc.co.uk/nature/life/Morelia_amethistina`;
  - 312: `http://en.openei.org/wiki/Special:ExportRDF/Colorado's_7th_congressional_district`;
  - 4370: `http://www.salon.com/writer/glenn_greenwald/`;
  - 335: `http://en.openei.org/wiki/Special:ExportRDF/Property:Name`;
  - 4376: `http://www.steelers.com/`;
- These clusters move over the map, and get almost perfectly merged together by 2013-04-07. At this point important BMUs in this area are:
  - 3991: `http://www.bbc.co.uk/nature/life/Myodes_glareolus`;
  - 313: `http://en.openei.org/wiki/Special:ExportRDF/Colorado`;
  - 4373/2: `http://www.sports-reference.com/olympics/athletes/br/heinz-brandt-1.html` / `http://www.snee.com/bob/foaf.rdf`;
  - 3819: `http://www.bbc.co.uk/nature/life/Gelada`;
  - 335: `http://en.openei.org/wiki/Special:ExportRDF/Property:Name`;
- 2013-8-11: The clusters have perfectly merged. The BMUs of the previous key epochs continue appearing as "shared" cluster centers;
- End of 2013: BMUs that did qualify as important in the previous epochs, but that did not make it as cluster centers, are not that obivously related to the cluster centers anymore. At this point, the BBC BMU and the OpenEI BMU seem to compete in importance in the unique remaining cluster;
- During 2014: BBC clusters get merged, so does the OpenEI by becoming indistinguishable within the BBC cluster;
- This tendency continues until the final data epoch in 2016-03-27.

[10] See the videos at `https://youtu.be/3nK3teAzzCM`, `https://youtu.be/XTcsv2i2Hlg`, and `https://youtu.be/-UKKIdIyKGA`



(a) Map for 2012-05-13.

(b) Map for 2013-04-07.

(c) Map for 2013-08-11.

(d) Map for 2013-12-29.

(e) Map for 2014-06-15.

(f) Map for 2016-03-27.

Fig. 1: Snapshots of key phases in the evolving map of the DyLDO dataset in the period from 2012-05-13 until 2016-03-27.

4.4 Discussion

The resulting maps described in Section 4.3 and the BMUs displayed therein seem to indicate that certain named graphs cluster together around some very specific datasets, namely:

- The BBC Nature data dataset. Some other BBC named graphs cluster together outside of the main cluster, but end up merging into one. The common vocabularies and predicates that can be found in these are RDF (`rdf:type`), RDFS (`rdfs:label`), the Wildlife Ontology, FOAF, and Dublin Core.
- The Open Energy Information wiki export as Linked Data. Like the BBC graphs, some other OpenEI named graphs are clustered separately before merging into one. The common vocabularies and predicates that are found in

these are RDF (`rdf:type`), Semantic MediaWiki terms, OWL (`owl:imports`, `owl:sameAs`), RDFS (`rdfs:label`), and custom OpenEI ontological terms.
– Olympics athletes Linked Data / FOAF file of Bob DuCharme, and more generally pages with personal details and FOAF data. In the olympics athletes pages, the used vocabularies are Facebook, the Open Graph Protocol, and XHTML; while in personal page's Linked Data the most common are RDFS (`rdfs:seeAlso`), RDF, and FOAF.

Besides the common usage of typical terms of RDF and RDFS, these datasets represent some significant actors of the LOD cloud: life sciences data and BBC datasets, encyclopedic knowledge coming from wikis, and RDF/RDFa data embedded in personal pages describing personal profiles. Apart from their fundamental differences (e.g. usage of FOAF, MediaWiki), subsequent SOMs tend to emphasize their common characteristics, grouping them. A fundamental question is, though, what makes these datasets so different from the surrounding, planar surface around the main clusters? We conjecture about plausible explanations for this. First, the higher number of instances using these terms might lead to a higher frequency in their sparse matrices. Second, the SOM maps reward the stability of these datasets versus others that changed their used vocabularies over time. Third, we observe that these datasets tend to reuse existing vocabularies, instead of minting their own (with few exceptions). Finally, all of them reuse very common terms that tend appear elsewhere in the LOD cloud. Anyhow, Figure 1 highlights publishers that kept their data accessible and that did not introduce new terminology nor removed it. We observe that this corroborates how systematic the DyLDO crawls were.

5 Conclusions and Future Work

In this paper we studied the requirements for visualizing large amounts of Linked Data that evolve over time, and proposed candidate solutions for fulfilling them. These candidate solutions were Somoclu, an efficient parallel implementation of self-organizing maps – an unsupervised dimensionality reduction technique; and DyLDO, a timestamped and systematic sample of the LOD cloud between 2012 and 2016. Our findings arose from an animation that told a story of stability, but that emphasized datasets with high availability, high vocabulary reuse, and priority for frequently used terms.

Many paths remain open, and we plan on extending this work in several ways. First, we will prepare the timestamped LOD with different assumptions beyond usage of predicates in named graphs. Second, we will treat different scalable datasets as both content repositories and fodders for semantic reasoning. Finally, we will integrate metadata of version control systems and data engineering workflows to better visualize, and understand, the semantic drift of Web data over time.

6 Acknowledgment

The second and third author acknowledge research funding by the European Commission Seventh Framework Programme under Grant Agreement Number FP7-601138 PERICLES.

References

1. Alexander Mäedche, Boris Motik, and Ljiljana Stojanovic: Managing multiple and distributed ontologies in the Semantic Web. The VLDB Journal — The International Journal on Very Large Data Bases 12(4), 286–300 (2003)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American 284(5), 34–43 (2001)
3. Dividino, R., Gottron, T., Scherp, A.: Strategies for efficiently keeping local linked open data caches up-to-date. In: Proceedings of ISWC-15, 14th International Semantic Web Conference. pp. 356–373 (2015)
4. Gonçalves, R.S., Parsia, B., Sattler, U.: Analysing Multiple Versions of an Ontology : A Study of the NCI Thesaurus. In: Proceedings of DL-11, 24th International Workshop on Description Logics. vol. 745 (2011)
5. Käfer, T., Abdelrahman, A., Umbrich, J., O'Byrne, P., Hogan, A.: Observing linked data dynamics. In: Proceedings of ESWC-13, 10th International Conference on the Semantic Web: Semantics and Big Data. pp. 213–227 (2013)
6. Kohonen, T.: Self-Organizing Maps. Springer (2001)
7. Leenheer, P.D., Mens, T.: Ontology evolution: State of the art and future directions. In: Ontology Management for the Semantic Web, Semantic Web Services, and Business Applications. Springer (2008)
8. Meroño-Peñuela, A.: Semantic Web for the Humanities. In: Proceedings of ESWC-13, 10th International Conference The Semantic Web: Semantics and Big Data. pp. 645–649 (2013)
9. Meroño-Peñuela, A.: Refining Statistical Data on the Web. Ph.D. thesis, Vrije Universiteit Amsterdam (2016)
10. Pesquita, C., Couto, F.M.: Predicting the Extension of Biomedical Ontologies. PLoS Computational Biology 8(9), e1002630 (2012)
11. Stavropoulos, T.G., Andreadis, S., Riga, M., Kontopoulos, E., Mitzias, P., Kompatsiaris, I.: A framework for measuring semantic drift in ontologies. In: Proceedings of SuCCESS-16, 1st Int. Workshop on Semantic Change & Evolving Semantics, co-located with the 12th European Conference on Semantics Systems (SEMANTiCS-16) (2016)
12. Stojanovic, L.: Methods and Tools for Ontology Evolution. Ph.D. thesis, University of Karlsruhe (2004)
13. Wang, S., Schlobach, S., Klein, M.C.A.: What Is Concept Drift and How to Measure It? In: Proceedings of EKAW-10, 17th International Conference on Knowledge Engineering and Management by the Masses. pp. 241–256 (2010)
14. Wittek, P., Darányi, S., Kontopoulos, E., Moysiadis, T., Kompatsiaris, I.: Monitoring term drift based on semantic consistency in an evolving vector field. In: Proceedings of IJCNN-15, International Joint Conference on Neural Networks (2015)
15. Wittek, P., Gao, S.C., Lim, I.S., Zhao, L.: Somoclu: An efficient parallel library for self-organizing maps. arXiv:1305.1422 (2015)