Towards assured data quality and validation by data certification

John P. McCrae
CITEC, Bielefeld University
Inspiration 1
Bielefeld, Germany
jmccrae@cit-ec.uni-bielefeld.de

Cord Wiljes
CITEC, Bielefeld University
Inspiration 1
Bielefeld, Germany
cwiljes@cit-ec.uni-bielefeld.de

Philipp Cimiano
CITEC, Bielefeld University
Inspiration 1
Bielefeld, Germany
cimiano@cit-ec.uni-bielefeld.de

ABSTRACT

Increasingly a large amount of data relevant to a wide variety of scientific domains is self-published by scientists on websites and this is proving to be an important resource for the replicability and further development of science. Much of this data is even made available as linked data. However, the self-publishing model provides no quality control on the data, and as such datasets frequently contain errors. We therefore consider an architecture of a system that enables the *certification* of data (both linked and otherwise) by a web service and the sharing of this certification on the web, and contemplate why this may improve data quality.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

science, data, quality

Keywords

data quality, data sharing, open science, validation service

1. INTRODUCTION

It has been widely acknowledged across the sciences that the publishing of data generated or required for an experiment is a crucial step towards the replicability of experiments or analyses [10]. However, it is also the case that most data is of poor quality and plagued by basic data errors [1, 11]. In this paper we tackle an aspect of data quality we refer to as the "readiness for use", by which is meant whether the data can be directly applied, rather than if its content is actually useful for a specific application. Errors such as these can easily be detected by validation in a manner that does not need to know the domain or the intended application of the data. Such errors not only make the data fundamentally harder to use but also mean that anyone consuming the dataset must first correct any existing data errors, possibly making unwarranted assumptions about the data, thus potentially leading to unintended modifications of the data. Much of this is due to the fact that for many small datasets there is no sufficient institutional support for the publication of data, leading to many datasets containing formal errors, such as incorrectly escaped characters. It is our belief that many scientists who self-publish datasets do not make such errors out of intention or indifference, but instead out of a lack of support in validating services. To this end we propose a simple, general, extensible web service to provide syntactic and semantic validation of data in, initially, XML and RDF, which can be extended to a wider range of data formats.

Finally, a second key goal is to provide *continuous validation* of the resource in that we continue to check the validity of resources periodically after they are published. It is in fact a common problem that resources and data cease to be available after the end of the funding period, and as such the data generated during this project become lost. It is also quite common for URLs to be changed for technical reasons without a redirect from the old URL to be implemented. For example, in a study of MEDLINE papers [12], it was found that 37% of URLs quoted in papers had become unavailable or were only intermittently available after publication, although it was unclear how many of these URLs referred to datasets.

The architecture of such a system has several clear design goals in order to cope with such a wide range of potential resources. The architecture should fulfill the following requirements:

Extensibility: There are a wide range of data formats in use in scientific work and as such we should be able to grow and extend to a wide range of data sets that are available on the web.

Efficiency: The system should be able to process potentially very large datasets in a reasonable amount of time. For that reason, validation algorithms that have a linear time complexity in the input size are to be preferred.

Tiered Architecture: It should be possible to follow deeper validation layers, such that we can validate data as to

LDQ 2014, 1st Workshop on Linked Data Quality; Editors: Magnus Knuth, Dimitris Kontokostas, and Harald Sack Sept. 2, 2014, Leipzig, Germany.

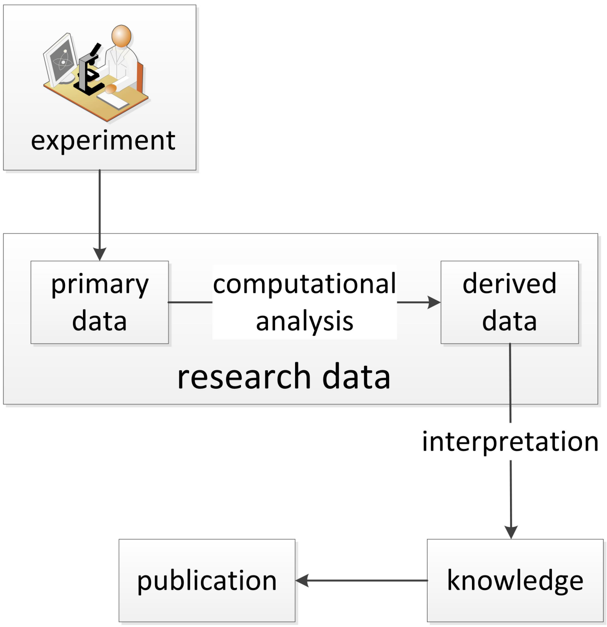


Figure 1: Research data in the scientific discovery process

> whether it is available on the web and whether it uses a standard and open format. Further, in the case that the data uses a valid RDF vocabulary, we can check whether it conforms to RDFS/OWL schema/ontology that the data claims to adhere to.

Such an architecture should allow us to quickly build an extensible service that allows new data formats and models to be handled and validated.

2. MOTIVATION

The Open Science movement advocates sharing the data that scientific results are based on [9]. Open data publication is expected to improve the integrity and efficiency of science. Errors and fraud will be easier to detect and valuable research data can be re-used by other scientists for their own research questions. Therefore, scientific journals and research funding agencies worldwide have been instituting policies for data sharing.

Good scientific practice calls for research to be reproducible, i.e. other researchers must be able to test the data as well as the analysis procedures. The growing number and diversity of digital research data and the strong increase in importance of computational methods in all empirical sciences have created hurdles for this ideal. Whereas in the past reproducibility in the scientific research process (Fig. 1) was mainly concerned with reproducing the experimental result in recent years it has become increasingly difficult to ensure the reproducibility of the computational analysis of research [3]. Therefore, a new "culture of reproducibility for computational science" [10] is needed.

For data to be useful it has to be of high quality, so additional efforts will be necessary to test and ensure data quality. Standards and best practices for data publication need to be defined. Building on the proven workflows for quality assessment in science we propose a combination of tool-assisted automated quality evaluation, complemented by a social, peer-reviewing based approach.

3. TARGET DATASETS

In general, we require that there are three main conditions on datasets that are necessary in order to build a service for the validation of datasets. Firstly, we would require that the dataset is *open*. In this case we do not require that the license is necessarily fully open, such as using a CC-BY[1] license, but rather this requirement states that we can access the datasets systematically by downloading them on the web, without the impediment of authentication systems or such like. Secondly, it is important that the dataset is a *single file*, as we wish to download the dataset without the user having to fill in complex metadata to describe how we may access individual files. We see no conceivable use case where a dataset cannot be combined into a single file by archiving or a similar method. Finally, we require that the dataset uses a *standard format*, that is a format that is open and is standardized by some standardization body. These requirements are similar to the 3rd star of the "5 Star Open Data Model" [2]. The advantage of these requirements is that we do not require complex metadata to describe a dataset but instead require only a download URL, which is easy to work with.

4. ARCHITECTURE

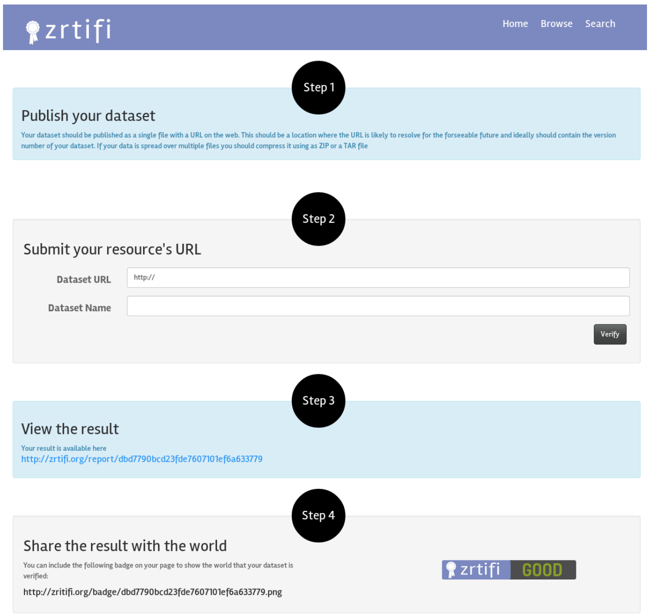
The certification system we propose in this paper takes the form of a very simple web service in which we take as input a single URL and then assign a local identifier (also a URL based on the MD5 hash of the external URL) to the dataset where we can make the results of the process available by means of linked data. As such the service is based around simple RESTful principles allowing a single URL to be posted to the service and a the resulting report URL returned by means of an HTTP redirect. Dereferencing the returned URL will give the current status of the resource as an RDF document based on the DCAT vocabulary [7] and the DataID scheme [2].

4.1 User interaction

A key goal of a web service is to engage with a wide range of data publishers including many who may not be familiar with web services and RESTful principles. As such, we acknowledge that it is important to enable the usage of the service by a wide range of users. Thus, we provide a simple form based interaction explaining to the user how to use the web service. Furthermore, they get to see the report URL immediately, which is based on a hash function and calculated in the browser.

Of most importance, however is the final step, where a certificate is provided which users can include on their own website next to the download link. This certificate will dynamically display the dataset's current evaluation as an iconic image, which will contain a brief summary of the dataset in terms of badges or stars awarded to the dataset based on the

[1] `https://creativecommons.org/licenses/by/4.0/`
[2] `http://www.w3.org/DesignIssues/LinkedData.html`



Publish your dataset

Your dataset should be published as a single file with a URL on the web. This should be a location where the URL is likely to resolve for the forseeable future and ideally should contain the version number of your dataset. If your data is spread over multiple files you should compress it using as ZIP or a TAR file

Submit your resource's URL

Dataset URL: http://

Dataset Name:

View the result

Your result is available here
http://zrtifi.org/report/dbd7790bcd23fde7607101ef6a633779

Share the result with the world

You can include the following badge on your page to show the world that your dataset is verified:

http://zritifi.org/badge/dbd7790bcd23fde7607101ef6a633779.png

Figure 2: A mock-up of the user page for the certification service

validation. This image will be provided directly at a URL derived from the dataset by means of a MD5 hash and will thus be up-to-date with current evaluations, and firmly tied to that URL encouraging data providers not to change URL without providing a URL forwarding mechanism.

Warning This URL is invalid, has not yet been analysed or the data set has not been available for more than three months

Bronze star It is possible to download this URL

Silver star It is possible to download this URL, extract it if necessary, and the data contains syntactically valid RDF or XML.

Gold star As silver, but deeper semantic validation (discussed below) was also successful.

Linked data star The data is valid and contains external links.

It is important to stress that the linked data star is not awarded for simply using RDF, but instead for having at least 50 triples[3] that refer to entities hosted on some other domain, where the domain of the dataset is assumed to be the same as its download URL.

These stars are included as part of the badge that the user can display on the website and as such allow external users to easily verify the quality of the downloaded dataset. These badges, which take the form of a custom generated PNG image, allow the data publisher to show the quality of their data[4], and assure the user of the quality of the data. This image's URL is related to the more detailed report and so it is easy to verify that it refers to the published dataset. Furthermore, by issuing a separate star for linking the dataset, we believe that this will be an enticement for data providers to follow linked data principles and thus move towards 5 star data as defined by Heath and Bizer [5].

4.2 Validation architecture

As our goal is to handle datasets which are both very large and potentially very diverse, the calculation of the validation system is far from a trivial implementation. To this end, we require that the validation itself follows specific requirements. The most important of these requirements are as follows:

- The service will not permanently store any data, both for practical reasons and to ensure that we do not violate any licenses. This means the service will not be able to act as a back-up or an alternative source of any of these data services. As such the service is not intended to replace the use of a DOI to provide a fix identifier for the data.
- The steps should be able to process the dataset in a single pass, without either using significant memory or requiring the creation of a large database. This requirement stops an execution of the validation from monopolizing the resources on the server.
- It should be possible to add new steps without significant modification to the system. This will enable not only us but also outside collaborators to contribute new validation steps, and as such we will make the source code available on the web and accept appropriate extensions.

The architecture of the system is illustrated in Figure 3. In this we see that the basic services start of with the download step, which as its name suggests obtains a copy of the resource by HTTP(S). The next step, which we call the *format sniffer*, attempts to deduce the format of the file. It does this by looking at the file name (extension), the HTTP headers and the first 1KB of the file. If the file is found to be an archive of some form then we extract it and apply the format sniffer to each extracted file. We also note that the format sniffer is extensible by means of *dependency injection* [8], allowing external contributors to easily add new formats.

Then the systems applies a format specific validator, such as a SAX parser for XML, or the *Rapper*[5] tool for RDF documents. Each of these services are implemented as a single command and are extended to return an RDF document. This RDF document contains the result of the execution (success, failure, internal error), any potential next steps to run in the chain and any extra annotations to be added

[3] Following `http://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/DataSets/CKANmetainformation`

[4] This is similar to the use of build status images used by continuous integration servers, such as by Travis CI

[5] `http://librdf.org/raptor/rapper.html`

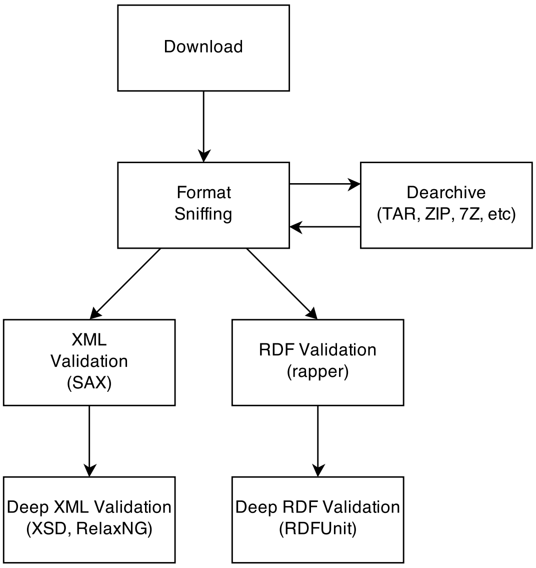

Download
Format Sniffing
Dearchive (TAR, ZIP, 7Z, etc)
XML Validation (SAX)
RDF Validation (rapper)
Deep XML Validation (XSD, RelaxNG)
Deep RDF Validation (RDFUnit)

Figure 3: The initial set of back-end validation services

to the report. For example, if the XML syntax validator finds a link to an XML document type definition (DTD) or schema description (XSD) then the service may indicate that validation according to the schema is the next step in the chain, which may have already been carried out by the SAX parser. Furthermore, the steps may yield additional output. For example, Rapper produces the number of triples and this is the added to the report using the VoID vocabulary[6]. Finally, we apply deeper tests to the RDF using the RDFUnit [6] framework, which checks whether the dataset conforms to the constraints defined by its ontological constraints. This framework is based on SPARQL and works on a principal of checking whether certain queries produce results as intended.

4.3 Continuous validation

While datasets are frequently of good quality when released, one of the key concerns in data quality is that eventually these datasets become unavailable or the URL they are published at changes. As such, our service plans to not only do initial validation but also to provide continuous validation. To this extent we will access the URL by means of a header-only-request (falling back to a `GET` for servers that do not support `HEAD`). Then by analysing the return status, especially the `Last-Modified` header, we can deduce if a resource is likely to have changed. In such cases we can re-run the full validation chain. If a resource fails over a fixed time period we will mark it as not downloadable.

5. CONCLUSION

In this paper we have presented the architecture of a system that aims to help with the quality of data and in particular linked data as self-published by scientists and other professionals on the web. This system works by means of certifying that datasets follow not only simple syntactic constraints of RDF and XML, but also deeper semantic conditions as defined by the schema. The system is currently under development and we expect to release the prototype version briefly after publication of this article. While it is clear that this service cannot guarantee that a dataset is fit for use in a given application, it can guarantee developers that the dataset is ready to be applied, avoiding the “tedious process of data wrangling” [4] by ensuring that formats are valid and encouraging the use of data semantics. We hope that by providing an easy-to-use interface, without requiring significant metadata, this service can play a key role in improving data quality and enabling replicability of experiments across all computational sciences.

Acknowledgments

This work is supported by the Cognitive Interaction Technology, Center of Excellence (CITEC) and the LIDER Project under the European Seventh Framework Program grant number 610782.

6. REFERENCES

[1] S. Bechhofer and R. Volz. Patching syntax in OWL ontologies. In *The Semantic Web – ISWC 2004*, pages 668–682. Springer, 2004.

[2] M. Brümmer, C. Baron, I. Ermilov, M. Freudenberg, D. Kontokostas, and S. Hellmann. DataID: Towards semantically rich metadata for complex datasets. In *Proceedings of the 10th International Conference on Semantic Systems*, 2014.

[3] D. L. Donoho, A. Maleki, I. U. Rahman, M. Shahram, and V. Stodden. Reproducible research in computational harmonic analysis. *Computing in Science & Engineering*, 11(1):8–18, 2009.

[4] P. J. Guo, S. Kandel, J. M. Hellerstein, and J. Heer. Proactive wrangling: mixed-initiative end-user programming of data transformation scripts. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 65–74, 2011.

[5] T. Heath and C. Bizer. Linked data: Evolving the web into a global data space. *Synthesis lectures on the semantic web: theory and technology*, 1(1):1–136, 2011.

[6] D. Kontokostas, P. Westphal, S. Auer, S. Hellmann, J. Lehmann, R. Cornelissen, and A. Zaveri. Test-driven evaluation of linked data quality. In *Proceedings of the 23rd international conference on World Wide Web*, pages 747–758, 2014.

[7] F. Maali, J. Erickson, and P. Archer. Data catalog vocabulary (DCAT). *W3C Working Draft*, 2012.

[8] R. C. Martin. The dependency inversion principle. *C++ Report*, 8(6):61–66, 1996.

[9] P. Murray-Rust, C. Neylon, R. Pollock, and J. Wilbanks. Panton principles: principles for open data in science. *Panton Principles*, 2010.

[10] R. D. Peng. Reproducible research in computational science. *Science*, 334(6060):1226, 2011.

[11] E. Rahm and H. H. Do. Data cleaning: Problems and current approaches. *IEEE Data Eng. Bull.*, 23(4):3–13, 2000.

[12] J. D. Wren. 404 not found: the stability and persistence of urls published in medline. *Bioinformatics*, 20(5):668–672, 2004.

[6] `http://www.w3.org/TR/void/`