Ya2ro: A tool for creating Research Objects from minimum metadata

Floriana Antonia Pavel[1] and Daniel Garijo[1[0000−0003−0454−7145]⋆]

Ontology Engineering Group, Universidad Politécnica de Madrid
antonia.pavel@alumnos.upm.es, daniel.garijo@upm.es

Abstract. Research Objects (ROs) have been proposed as a packaging mechanism to aggregate research outputs of scientific investigations and capture their context and metadata in a machine-readable manner. However, creating ROs (and collecting the respective metadata of their constituent resources) is still a time-consuming task. In this demo we present ya2ro, a tool designed to ease the creation of Research Objects following the RO-Crate specification. Given an input file with external resources available on the Web (datasets, software, publications and people), ya2ro will retrieve their metadata descriptions (if available), creating an aggregated RO-Crate available both in human-readable manner (HTML) and machine-readable manner (JSON-LD).

Keywords: Research Object · metadata · dataset · software

1 Introduction

Over the last decade, Research Objects (ROs) [11] have been proposed as a means for packaging the context and artifacts associated with a research investigation, in domains ranging from Computational Biology [4] to Geosciences [9]. ROs enable researchers aggregating diverse research outputs such as datasets, code, workflows or existing publications into *digital objects*, capturing their context, relationships and metadata in a machine-readable manner.

In order to ease creating ROs, the community has proposed programmatic tools like ro-crate-py [2], web services like Describo[1] and platforms like RO-Hub [9], which generate ROs according to the RO-Crate specification [11]. However, creating ROs is still largely a manual process that takes significant time, especially when aggregating external resource metadata that are described elsewhere in the Web with different identifiers (DOIs, URLs, ORCIDs, etc.) and accessible through heterogeneous APIs (REST, SPARQL endpoints, etc.).

⋆ This work has been supported by the Madrid Government (Comunidad de Madrid-Spain) under the Multiannual Agreement with Universidad Politécnica de Madrid (UPM) in the line Support for R&D projects for Beatriz Galindo researchers, in the context of the V PRICIT (Regional Programme of Research and Technological Innovation) and the call Research Grants for Young Investigators from UPM

[1] https://github.com/Arkisto-Platform/describo

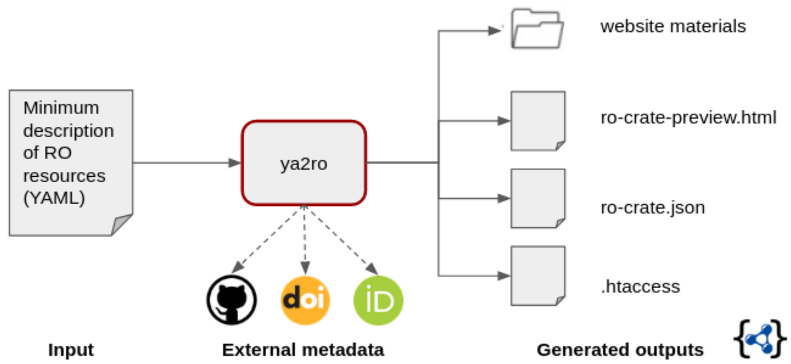

Fig. 1. Main ya2ro workflow: Given a YAML file with identifiers (DOIs, ORCIDs, code repository URLs) and basic descriptions (summary, title) as input to the tool, ya2ro will collect existing metadata from external APIs and generate a Research Object.

In this demo we present ya2ro, a tool designed to ease describing resources by collecting existing metadata from datasets, software, publications and persons when creating a RO. As a result, ya2ro generates a customizable HTML representation of the RO, its equivalent machine-readable JSON file and the means to perform content negotiation.

2 Ya2ro: Creating ROs from YAML files

The design of ya2ro was driven by three main objectives: 1) reducing the technical needs to adopt ROs by users non familiar with JSON-LD [7] or RDF [1], 2) automatically enriching ROs with metadata from external resources referenced in them; and 3) automatically creating human-readable descriptions (i.e., a website) from the enriched results.

Figure 1 shows an overview of the steps followed by ya2ro to create a Research Object as an RO-Crate. The tool expects a YAML file as input, which contains basic descriptions of the RO (e.g., title, summary), and a list of resources organized in the following categories: 1) Datasets (i.e., data used or generated by the research described in the RO), 2) Software (i.e., main tools developed to process or generate results), 3) Bibliography (i.e., key publications used for the research, or where the research is published in) and 4) RO Authors and their role (coding, supervising, experiment design, etc.).

Listing 1 shows a sample YAML file used by ya2ro. Each resource can be described by using an external identifier (DOIs for datasets and publications, ORCID for authors, code repository URLs for software), or by adding basic metadata (name, description and license). Existing documentation shows all the supported metadata fields by our tool. [2]

[2] https://github.com/oeg-upm/ya2ro/blob/main/Documentation.md

Listing 1: sample YAML file for creating a Research Object.

```
type: "paper"
summary: "Summary of the paper (abstract)"
title: "Paper title"
datasets:
  - Dataset DOI
software:
  - Github URL
bibliography:
  - Paper DOI
  - "Citation" (alternative to DOI)
authors:
  - orcid: first author ORCID
    role: role of the author
  - name: second author name (alternative to ORCID)
    description: author brief bio
    position: organization of the author
    role: role of the author
```

Ya2ro resolves the identifiers provided by users and adds the corresponding metadata in the RO. For Dataset DOIs, ya2ro extracts a title, description and license if a JSON-LD representation is available. For publication DOIs, authors and title are retrieved in a similar fashion (retrieving a bibtex serialization if available). For authors, ya2ro uses the ORCID API[3] to retrieve name, position and a brief description. Finally, software projects are described using existing software metadata extraction [8] [6] and code analysis [3] tools, integrating their results into the RO. All metadata fields are added in the RO following the RO-Crate specification, by aligning them against Schema.org [5] terms.

Figure 2 shows a fragment of the HTML results generated by ya2ro, showcasing the automatically retrieved software metadata in grey. In this case, a short software description is shown, together with small badges at the bottom of grey area which have additional information on license, how to cite the software component, its status and acknowledgement information. The HTML page[4] also provides a link to the RO-Crate JSON-LD file (point 5 in Figure 2) and a link to an explanation of the results (point 6) which details the YAML template used. The tool will also validate the metadata found, asking users to complete minimum information about their resources (title, description) when no metadata is found. Ya2ro is open source (Apache 2.0 license) and available online in GitHub[5] and pypi[6] [10].

[3] https://info.orcid.org/documentation/features/public-api/
[4] Figure example available at https://w3id.org/dgarijo/ro/sepln2022
[5] https://github.com/oeg-upm/ya2ro
[6] https://pypi.org/project/ya2ro/

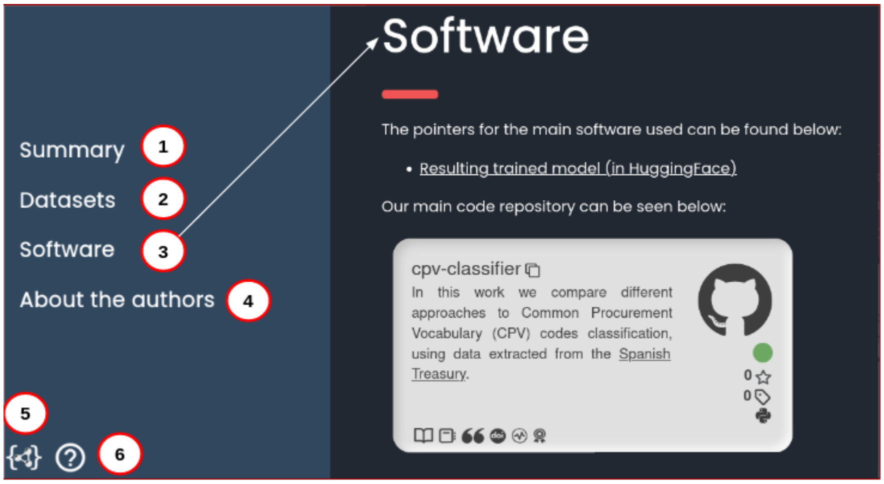

Fig. 2. RO-crate HTML preview generated by ya2ro. The HTML contains a summary (1), dataset (2) and software (3) metadata, and author information (4). A link to the JSON-LD representation (5) and an explanation of each field (6) are also included.

3 Conclusions and Future Work

This demo presents ya2ro, a tool designed to ease the creation of ROs by researchers with little experience in semantic technologies. We are currently expanding ya2ro to support collecting metadata from other types of artifacts, like workflows or ontologies. We are also exploring new means to capture relationships within the RO itself, e.g., data and software dependencies.

References

1. Cyganiak, R., Lanthaler, M., Wood, D.: RDF 1.1 concepts and abstract syntax. W3C recommendation, W3C (Feb 2014), https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/
2. De Geest, P., Droesbeke, B., Eguinoa, I., Gaignard, A., Huber, S., Leo, S., Pireddu, L., Rodríguez-Navas, L., Sirvent, R., Soiland-Reyes, S.: ro-crate-py (May 2022). https://doi.org/10.5281/zenodo.6594974
3. Filgueira, R., Garijo, D.: Inspect4py: A knowledge extraction framework for python code repositories. In: IEEE/ACM 19th International Conference on Mining Software Repositories, MSR 2022, Pittsburgh, PA, USA, May 23-24, 2022. pp. 232–236. IEEE (2022). https://doi.org/10.1145/3524842.3528497
4. Goble, C., Soiland-Reyes, S., Bacall, F., Owen, S., Williams, A., Eguinoa, I., Droesbeke, B., Leo, S., Pireddu, L., Rodríguez-Navas, L., Fernández, J.M., Capella-Gutierrez, S., Ménager, H., Grüning, B., Serrano-Solano, B., Ewels, P., Coppens, F.: Implementing FAIR Digital Objects in the EOSC-Life Workflow Collaboratory (Mar 2021), https://doi.org/10.5281/zenodo.4605654, white paper

5. Guha, R.V., Brickley, D., Macbeth, S.: Schema. org: evolution of structured data on the web. Communications of the ACM **59**(2), 44–51 (2016). https://doi.org/10.1145/2844544
6. Kelley, A., Garijo, D.: A Framework for Creating Knowledge Graphs of Scientific Software Metadata. Quantitative Science Studies pp. 1–37 (11 2021). https://doi.org/10.1162/qss_a_00167
7. Longley, D., Champin, P.A., Kellogg, G.: JSON-ld 1.1. W3C recommendation, W3C (Jul 2020), https://www.w3.org/TR/2020/REC-json-ld11-20200716/
8. Mao, A., Garijo, D., Fakhraei, S.: Somef: A framework for capturing scientific software metadata from its documentation. In: 2019 IEEE International Conference on Big Data (Big Data). pp. 3032–3037 (2019). https://doi.org/10.1109/BigData47090.2019.9006447
9. Palma, R., Garcia-Silva, A., Gomez-Perez, J.M., Krystek, M.: A research object-based toolkit to support the earth science research lifecycle. In: 2018 IEEE 14th International Conference on e-Science (e-Science). pp. 50–57. IEEE (2018). https://doi.org/10.1109/eScience.2018.00020
10. Pavel, A., Garijo, D., Str3am786: oeg-upm/ya2ro: ya2ro 0.0.4 (Apr 2023). https://doi.org/10.5281/zenodo.7803628
11. Soiland-Reyes, S., Sefton, P., Crosas, M., Castro, L.J., Coppens, F., Fernández, J.M., Garijo, D., Grüning, B., La Rosa, M., Leo, S., et al.: Packaging research artefacts with ro-crate. Data Science **Pre-press**, 1–42 (2022). https://doi.org/https://doi.org/10.3233/DS-210053