BioHackrXiv

Linking PubDictionaries with UniBioDicts to support Community Curation

John Zobolas[1, *], **Jin-Dong Kim**[2, *], **Martin Kuiper**[1], **and Steven Vercruysse**[3]

1 Department of Biology, Norwegian University of Science and Technology (NTNU), Trondheim, Norway **2** Database Center for Life Science (DBCLS), Tokyo, Japan **3** Independent Scientist, Trondheim, Norway ***** Shared first authorship

BioHackathon series:
BioHackathon Europe 2020
Virtual conference 2020

Submitted: 01 Jan 2021

Published by BioHackrXiv.org

Abstract

One of the many challenges that biocurators face, is the continuous evolution of ontologies and controlled vocabularies and their lack of coverage of biological concepts. To help biocurators annotate new information that cannot yet be covered with terms from authoritative resources, we produced an update of **PubDictionaries**: a resource of publicly editable, simple-structured dictionaries, accessible through a dedicated REST API. PubDictionaries was equipped with both an enhanced API and a new software client that connects it to the **Unified Biological Dictionaries** (UBDs) uniform data exchange format. This client enables efficient search and retrieval of *ad hoc* created terms, and easy integration with tools that further support the curator's specific annotation tasks. A demo that combines the Visual Syntax Method (VSM) interface for general-purpose knowledge formalization, with this new PubDictionaries-powered UBD client, shows it is now easy to incorporate the user-created PubDictionaries terminologies into biocuration tools.

Introduction

The curation of biological information is met with several challenges today. The constant refactoring of ontologies, nomenclature and identifiers, as well as the discovery of new types of information and new uses thereof, makes the life of knowledge curators difficult, especially in the highly diverse domain of biology. For example, expert curators who rely on specialized software tools in the annotation process might come across a new concept or feature that does not exist within an ontology or vocabulary that their annotation tool connects to. Similar difficulties are faced by non-expert curators. Some biologists want to create a project-specific knowledge resource in a biological niche that is only minimally or not yet covered by existing controlled vocabularies. Under time pressure of project milestones, these biologists may not immediately have the resources to organize a multilateral effort to standardize the terminology in their field. Instead they typically resort to creating 'private' vocabularies within their projects, that may later serve as a first step toward larger-scale coordination (Hartmann et al., 2019).

In cases where there is an abundance of resources for controlled vocabularies, ontologies and identifiers, it may still be challenging to coordinate access to these many necessary resources for dedicated annotation endeavours. Alternatively, in cases where no proper controlled vocabulary would exist, the results from all the work that goes into creating new vocabularies will remain largely isolated from general use, if no term sharing mechanism is available. Existing tools would then also be unable to access these newly created terms and for example serve them to curators in an autocomplete panel, to make their task easier and less error-prone.

These problems are addressed by two complementary initiatives. On the one hand, **Unified Biological Dictionaries** (**UniBioDicts** or **UBDs**) (Zobolas, Touré, Kuiper, & Vercruysse, 2020) is a set of software packages that offers a unified, single access point to biological terminology resources, and is available to be plugged into any curation platform currently in operation (for example the general-purpose curation interface 'vsm-box' (Vercruysse et al., 2020), based on VSM (Vercruysse & Kuiper, 2020), is out-of-the-box compatible with UBDs). On the other hand, **PubDictionaries** (Kim et al., 2019) is a public repository of dictionaries where users can create and immediately share their own dictionaries based on a simple data format consisting of a term and an identifier.

During the ELIXIR BioHackathon 2020, we updated and improved the PubDictionaries API as well as developed a new UBD client that directly communicates with that API. These implementation efforts constitute a significant step towards the unification of some of the most important data resources across all biological domains. The addition of PubDictionaries to the list of UniBioDicts-interoperable resources now provides a uniform way to search and autocomplete terms from all these community-created dictionaries as well. Such functionality enables the easier integration of PubDictionaries in any curation tool that may have a need for their terms and for such a term suggestion feature.

Results

In the following two subsections, we briefly summarize the results of the hackathon efforts, grouped into two categories:

- Updating the PubDictionaries REST API
- Creating a new UBD client library to access the above API

PubDictionaries REST API

PubDictionaries is a public repository of dictionaries, where each dictionary is a collection of **labels** (human-friendly **terms**) and **identifiers** (unambiguous **IDs**, used by computers). Each label + ID pair is called an **entry**. Multiple entries can have the same ID (for synonymous labels) and the same label can occur in multiple entries (for ambiguous ones). Users can create their own dictionaries and add entries to them via the web-interface. The dictionaries can be used to annotate any piece of text via the PubAnnotation ecosystem (Kim et al., 2019) or to simply lookup terms, and both these services are supported by a RESTful API (Kim, 2020). All the API responses are structured as JSON objects. Prior to the BioHackathon event, the REST service provided the following main endpoints:

1. `find_ids`: given some specific terms and dictionary names, this endpoint returns the corresponding IDs that approximately match the terms in these dictionaries. Example: https://pubdictionaries.org/find_ids.json?labels=TP53&dictionaries=human-UniProt
2. `prefix_completion`/`substring_completion`: given a term (or partial term string), these endpoints search for prefix (respectively substring) matches in a specified dictionary and return the corresponding entries. Note that only a first page of results was returned with at most 15 entries, prior to the hackathon efforts. Example: https://pubdictionaries.org/dictionaries/human-UniProt/prefix_completion?term=p53
3. `text_annotation`: given a piece of text and dictionary names, this endpoint returns the result of annotation to the text using the dictionaries. The annotation is performed based on computation of string similarity between dictionary entries and expressions in the text. Example: https://pubdictionaries.org/text_annotation.json?dictionary=human-UniProt&text=The%20tumor%20suppressor%20p53%20(TP53)%20is%20the%20most%20frequently%20mutated%20human%20gene

The following REST Service endpoints were added during the BioHackathon:

1. `dictionaries` endpoint: returns information about a specific dictionary, such as its id, name, a text description, the number of entries it has, etc. Example: https://pubdictionaries.org/dictionaries/human-UniProt.json, where `human-UniProt` can be any existing dictionary name.
2. `entries` endpoint: returns all entries of a specific dictionary, paginated and sorted by label. Example: https://pubdictionaries.org/dictionaries/human-UniProt/entries.json?page=1&per_page=15 Note that the users (software developers) can explicitly configure how the result should be paginated, i.e. how many entries of a dictionary should be included in one 'result-page', and what page they want to get results back from.
3. `find_terms` endpoint: this is the complement of the `find_ids` endpoint in the sense that it returns a list of terms and dictionary names that match the given IDs. The result is first sorted by ID and then by dictionary name. If no dictionary name is given to this endpoint, then it searches for the given IDs in all dictionaries. Example: https://pubdictionaries.org/find_terms.json?dictionaries=&ids=https://www.uniprot.org/uniprot/P04637
4. `mixed_completion` endpoint: a combined and updated version of the `prefix_completion` and `substring_completion` endpoints. For a given term (or partial term string) and a specified dictionary it returns a list of entries, putting the prefix completions in the top half and the substring completions in the bottom half, while pruning any possible common entries. In addition, this endpoint supports pagination which is a direct result of extending the prefix and substring endpoints to support this feature as well. Example: https://pubdictionaries.org/dictionaries/human-UniProt/mixed_completion?term=p53&page=2&per_page=3

Additional work on the PubDictionaries server-side included the support of create (via the HTTP POST method) and delete operations of a specific dictionary, given certified user credentials.

Lastly, the error handling was harmonized across all REST URL endpoints. In particular, when a user searches for a non-existent dictionary name, the PubDictionaries server returns a proper HTTP response status code, 400 (Bad Request), together with a JSON-formatted description as follows: `{ "message": "Unknown dictionary: <name>." }`. For example, all the following URL links return such a response object:

- https://pubdictionaries.org/dictionaries/non-existent-dictionary-name.json
- https://pubdictionaries.org/find_terms.json?dictionaries=non-existent-dictionary-name&ids=id1,id2
- https://pubdictionaries.org/dictionaries/non-existent-dictionary-name/entries.json

UBD Client for PubDictionaries

UBDs are a set of software packages that provide a unified query-interface for accessing the online API services of key biological vocabulary-data providers (Zobolas et al., 2020). The main feature of UBDs is their string-search functionality, which returns for a given label (or partial label) a list of matching term, identifier and metadata units from databases (e.g. UniProt (The UniProt Consortium, 2019)), controlled vocabularies (e.g. PSI-MI), and ontologies (e.g. Gene Ontology, via BioPortal (Whetzel et al., 2011)). This feature makes UBDs ideal for enabling autocomplete support in user-interface components that serve terms to curators from disparate resources, thus allowing the more efficient annotation of information.

Our work in the ELIXIR BioHackathon 2020 included the creation of a new UBD client (vsm-pubdictionaries) that utilizes the updated PubDictionaries API in order to solve a long-standing problem in the biocurator community: how can *ad hoc*, project-specific terms and new information be effectively annotated with, and served via a curation platform, without the need to first negotiate the storage, update and reconciliation of that information with a third party, e.g. a database or ontology provider? Our client software addresses this problem

by presenting a mediator solution that can easily be plugged into current curation applications and serve *ad hoc* terms from PubDictionaries' public curator-created dictionaries.

Regarding the software client code, we wrote extensive documentation to delineate the mapping between the terms and IDs from PubDictionaries and the unified UBD format and how this is achieved via the updated PubDictionaries REST API endpoints, all in accordance with the UBDs' shared dictionary interface specification. We also enabled continuous integration support via GitHub Actions and wrote extensive tests (code coverage is at 95%), so as to deliver more reliable, fault-tolerant and easy-to-extend software. Moreover, the documentation includes two examples; one showcasing the search term functionality via `Node.js` and one indicating how to use the client library in a web-based environment, with an HTML file. Finally, the demo example (see Figure 1) that was presented during the last report session of the BioHackathon, demonstrates a simple use-case where a few public dictionaries were created and their terms served in a vsm-box curation interface (Vercruysse et al., 2020). Thus we show how straightforward the annotation of new information can be by means of the autocomplete functionality of the provided curation tool, and how this new knowledge can be connected with semantically aware annotations.

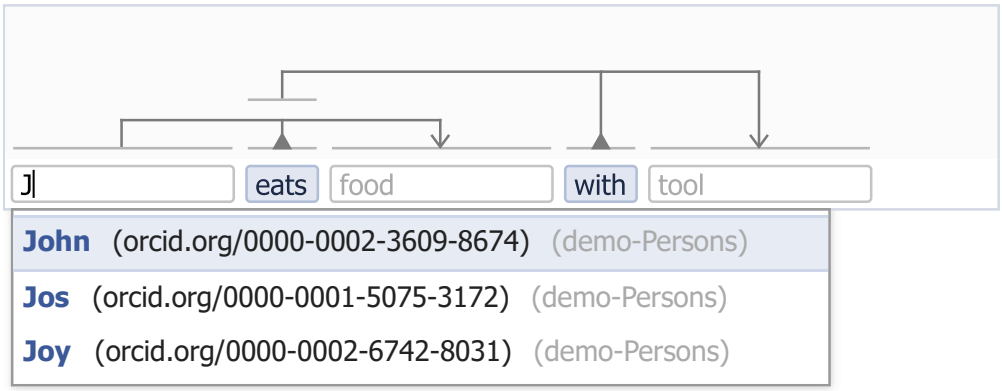

Figure 1: Demo example that uses a 'vsm-box' curation interface component, pre-filled with a VSM-template. An autocomplete panel appears while the user enters terms linked to identifiers. These term+ID pairs come from demo-dictionaries that we created at PubDictionaries, and are fetched through the new REST API-client described in this paper. Placeholders like 'food' and 'tool' indicate the kinds of dictionaries that specific fields of the template are connected with. On top, VSM-connectors formalize the structure and meaning of this knowledge unit.

Discussion

The advantages of a software package that connects with and queries any dictionary created in the PubDictionaries web-interface are multiple. This novel software enables annotation tools to use a common language and interface to link to information that is not yet available in standard databases. Note that the process of integrating new terms into standard resources can be time-consuming, so supporting communities of curators to create and utilize new terms that are at least publicly shared, in PubDictionaries, is a helpful first measure to tackle the problem of missing terms during ongoing curation work. Our software is a step towards achieving that goal, since it positions the community-manageable PubDictionaries into the mainstream of controlled vocabularies (CVs) and ontology resources. It fills the niche of new and *ad hoc* CVs that in turn may prompt new dedicated efforts to mature these CVs for consensus and expert maintenance.

Propelled by an ELIXIR BioHackathon event, our work underpins the goals of several of ELIXIR's activities, or so-called 'Platforms'. In the ELIXIR Data Platform, the drive to use, re-use and value life science data takes precedence. Our efforts exemplify how to achieve this by providing a scalable solution for curation platforms, especially ones that include support for annotation efforts that involve new information types. Furthermore, our main objective matches the goals of the ELIXIR Interoperability Platform: we offer a way to publicly-access and integrate new curated knowledge in a unified form, which enables new knowledge to be used by humans and machines alike, and to build knowledge systems that will aid future endeavors in understanding biological processes.

Future Work

Our future work includes updates on the PubDictionaries API to support the addition, update and deletion of dictionary entries, which is a functionality currently only available in the web-interface of PubDictionaries. Consequently, a further update on the UBD client will provide the necessary backbone to help build user interfaces, where curators would not even need to log in to the PubDictionaries website to create new dictionaries, and add, update or delete entries, but rather would be able to do that from within their own in-house curation tool. This functionality is currently not offered by any other biological data provider. Lastly, we expect that these updates, coupled with the search-string functionality provided by the PubDictionaries UBD client, will contribute in efforts to significantly increase the autonomy of biocurators and their potential for creating shareable annotations.

Links to software and documentation

- PubDictionaries API documentation: https://docs.pubdictionaries.org
- PubDictionaries source code: https://github.com/pubannotation/pubdictionaries
- UBD Client for PubDictionaries: https://github.com/UniBioDicts/vsm-pubdictionaries
- Demo example with vsm-box curation interface: https://github.com/UniBioDicts/vsm-pubdictionaries/blob/master/test/test_vsm_box_pubdictionaries.html

Acknowledgements

This work was carried out during the virtual Europe BioHackathon event that was organized by ELIXIR in November 2020. We would like to thank the organizers for the excellent management of such a large-scale, virtual event with 200+ participants and for creating the opportunity to meet, discuss, collaborate and share ideas and technologies with many new people.

References

Hartmann, N. B., Hüffer, T., Thompson, R. C., Hassellöv, M., Verschoor, A., Daugaard, A. E., Rist, S., et al. (2019). Are We Speaking the Same Language? Recommendations for a Definition and Categorization Framework for Plastic Debris. *Environmental Science and Technology*, *53*(3), 1039–1047. doi:10.1021/acs.est.8b05297

Kim, J.-D. (2020). PubDictionaries REST API documentation. Retrieved from https://docs.pubdictionaries.org/

Kim, J.-D., Wang, Y., Fujiwara, T., Okuda, S., Callahan, T. J., & Cohen, K. B. (2019). Open Agile text mining for bioinformatics: the PubAnnotation ecosystem. *Bioinformatics*, *35*(21), 4372–4380. doi:10.1093/bioinformatics/btz227

The UniProt Consortium. (2019). UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Research*, *47*(D1), D506–D515. doi:10.1093/nar/gky1049

Vercruysse, S., & Kuiper, M. (2020). Intuitive representation of computable knowledge. *Preprints*. doi:10.20944/preprints202007.0486.v2

Vercruysse, S., Zobolas, J., Touré, V., Andersen, M. K., & Kuiper, M. (2020). VSM-box: general-purpose interface for biocuration and knowledge representation. *Preprints*. doi:10.20944/preprints202007.0557.v1

Whetzel, P. L., Noy, N. F., Shah, N. H., Alexander, P. R., Nyulas, C., Tudorache, T., & Musen, M. A. (2011). BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic acids research*, *39*(Web Server issue), W541–5. doi:10.1093/nar/gkr469

Zobolas, J., Touré, V., Kuiper, M., & Vercruysse, S. (2020). UniBioDicts: Unified access to Biological Dictionaries. *Bioinformatics*. doi:10.1093/bioinformatics/btaa1065