doi:10.20944/preprints202007.0586.v1

UniBioDicts: Unified access to Biological Dictionaries

John Zobolas[1,*], Vasundra Touré[1], Martin Kuiper[1], and Steven Vercruysse[1]

[1]Department of Biology, Norwegian University of Science and Technology (NTNU), NO-7491 Trondheim, Norway

[*]To whom correspondence should be addressed.

Abstract

Summary: We present a set of software packages that provide uniform access to diverse biological vocabulary resources that are instrumental for current biocuration efforts and tools. The Unified Biological Dictionaries (**UniBioDicts or UBDs**) provide a single query-interface for accessing the online API services of leading biological data providers. Given a search string, UBDs return a list of matching term, identifier and metadata units from databases (e.g. UniProt), controlled vocabularies (e.g. PSI-MI), and ontologies (e.g. GO, via BioPortal). This can be coupled to for instance the 'vsm-autocomplete' module: an input field (user-interface component) that offers autocomplete lookup for these dictionaries. UBDs create a unified gateway for accessing life science concepts, helping curators find annotation terms across resources (based on descriptive metadata and unambiguous identifiers), and data users search and retrieve the right query terms.

Availability and implementation: The UBDs are available through *npmjs* (https://www.npmjs.com/search?q=vsm-dictionary) and the code is available in the GitHub organisation UniBioDicts (https://github.com/UniBioDicts) under the AGPL-3.0 license.

Contact: john.zobolas@ntnu.no, steven.vercruysse@ntnu.no

Supplementary information: Further information on VSM is available at https://vsmjs.github.io

1 Motivation

The plethora of ontology terms and biological entity identifiers (IDs) provides a vast resource for use in annotations (by curators) and in database queries (by life scientists and computers), but specifying and finding them requires extensive navigation through an intimidating number of web resources and look-up forms. A universal way to perform a comprehensive search of life science databases, ontologies and vocabularies, supported by an autocomplete function that allows users to choose from a list of candidate terms with defining metadata, will greatly streamline this process. In addition, it will help to eliminate errors that stem from typing these terms manually without autocomplete support or options for semantic input checking. Furthermore, a unified lookup utility makes terms from diverse vocabularies easy to place together into context-rich annotations. VSM for example (Vercruysse and Kuiper, 2020), a technology that allows the flexible annotation of virtually any type of contextual information, can take advantage of unified access to such a large diversity of terms, e.g. in applications like causalBuilder (Touré *et al.*, 2020). For these reasons, we set out to create a software suite that maps many of the diverse resources to a single data access and representation form.

2 Implementation

Each UBD module is an interface to an online server that provides ontology or controlled vocabulary data. A single dictionary module may provide access to one or several apparent 'sub-dictionaries'; e.g. the BioPortal UBD presents each of its many combined biological-domain ontologies as a distinct sub-dictionary. When a UBD receives a request for data, it makes a custom request to the associated server's API, and translates received data back into the format specified by the generic parent dictionary class. This data format is also recognized by a highly customizable autocomplete web-component (Vercruysse *et al.*, 2020).

2.1 Main data-types and methods

UniBioDicts works with three main data-object types:

1. `dictInfo`: this object holds information about one sub-dictionary of the data resource.

2. `entry`: this object represents all relevant information about a specific biological concept. It is the combination of a computer-processable ID, at least one human-friendly term (a word or word sequence), and various metadata. The combined metadata makes it possible to inform curators of what a concept represents and how its meaning differs from others. For example, the UniProt UBD returns the "tp53" concept via the standard properties: *id* (a URI, Uniform Resource ID: "https://www.uniprot.org/uniprot/P04637"), *terms* (a list: "P53_HUMAN", "Cellular tumor antigen p53", etc., with recommended name first, and synonyms next), *descr* (text description of the protein), *dictID* (URI for the resource: "https://www.uniprot.org"); and an extra set of *z* sub-properties for data specific to UniProt: *species* ("Homo Sapiens"), *genes* ("TP53", "P53"), etc.

3. `match`: this object combines one term-string (which may be a synonym, for one or several entries) with a specific `entry` that it represents. It is returned by UBDs' search-string query function. For example, querying the UniProt dictionary for "tumor antigen p53" returns among others the above entry object for "tp53", augmented with the property *str* ("P53_HUMAN").

Each UBD module offers the following methods to access a resource's data (along with options for filtering, sorting, and paging of results): `getDictInfos`, `getEntries` and `getEntryMatchesForString`, which return a list of `dictInfo`, `entry` and `match` objects, respectively. See online for the full specification of the parent interface vsm-dictionary and each UBD.

2.2 Additional features

Some additional features of UBDs are:

1. Several are optimized for curator use: their `match` object's *descr* and *str* can be tweaked to better reflect what biocurators are familiar with

in their annotation task; in particular for choosing the intended concept in an autocomplete list. For example, when the Ensembl UBD queries its server for “tp53”, it receives several gene concepts with the same name and description, but different species and gene-synonyms. So to provide a more informative description, the last three are combined into an optimized *descr*.

2. Identifiers (*id*, *dictID*) are formed as unambiguous, browsable URIs. This supports giving users clickable access to details about a returned concept to verify if it conveys the desired semantics for their annotation (McMurry *et al.*, 2017). Only the Noctua UBD returns CURIEs (Compact URIs) for use in Noctua-related applications.

3. UBDs have a future-proof design. For example, if a resource’s API offers additional information on a concept, then this extra data can be added in the `entry`’s *z*-object property, where it can be used to customize or augment what an autocomplete shows to the user.

3 Results

3.1 Implemented UBDs

The implemented UBDs map and unify the following biological resources and their respective APIs:

- BioPortal (Whetzel *et al.*, 2011), the largest repository of biomedical ontologies, using the BioPortal REST API (http://data.bioontology.org/documentation)
- PubMed MEDLINE database of biomedical literature, using the Entrez programming utilities (Sayers, 2020)
- Noctua Entity Ontology, using their Solr Web service (http://golr-aux.geneontology.io/)
- UniProt (The UniProt Consortium, 2019), using their REST API (https://www.uniprot.org/help/api)
- Ensembl (Zerbino *et al.*, 2018)

doi:10.20944/preprints202007.0586.v1

- Ensembl Genomes (Howe *et al.*, 2020)
- RNAcentral (The RNAcentral Consortium, 2018)
- Complex Portal (Meldal *et al.*, 2019)

Some dictionaries map to a single sub-dictionary, i.e. one term-list (e.g. UniProt), while others have several sub-dictionaries, i.e. a collection of terminologies (e.g. BioPortal's ontologies). The last four dictionaries each process a different data domain from the EBI Search API (Madeira *et al.*, 2019).

In addition, we provide a package that can combine several UBDs into one virtual dictionary implementing the same interface (vsm-dictionary-combiner), enabling the querying of multiple UBDs through one access point (e.g. an autocomplete component). An example of how this functions in practice can be observed in causalBuilder, a web interface tailored to the curation of causal interactions (Touré *et al.*, 2020).

3.2 Potential Users

The UBDs benefit three types of users:

1. Research software engineers who use UBDs as a meta-API. They can programmatically access multiple resources in a uniform way and avoid dealing with disparate APIs that all have different documentation, specifications, and data formats.
2. Software developers who build a project-specific curation tool. They can create input fields that offer autocomplete lookup in any set of UBDs and present matching terms and IDs in a selection panel. This is easily achieved by linking any dictionary to our vsm-autocomplete web-component (a reusable web-page element). UBDs can also be linked to a vsm-box to build curation applications, like causalBuilder.
3. Biocurators who use the above curation tools to find the terms they need. Autocomplete-based annotation allows biocurators to curate papers more quickly, conveniently, and precisely, without having to copy text and IDs from elsewhere (Jung *et al.*, 2009; Ward *et al.*, 2012).

4 Discussion

Although many of the leading resources provide individual support for finding appropriate identifiers, terms and definitions for biological entities and concepts, an overarching function that spans all resources was not yet available. Such a utility, providing real-time access to terminology from diverse biological subdomains through a unified interface, enables the development of tools that build upon the collective information residing in these disparate domains. A unified access to the wealth of descriptive information forms an essential enabling part of computational, semantic systems biology. Continuing in this spirit, we plan to build another UBD that connects with PubDictionaries (Kim *et al.*, 2019), and we invite future collaborators to join our UniBioDicts GitHub organisation and help build a growing collection of biological dictionaries. The currently developed packages cover a diverse range of web-services, API-technologies and associated data-types, providing concrete examples that facilitate the development of additional UBDs, or for that matter, any other domain dictionaries that may need to access online databases or ontologies for curation.

In the process of building the UBDs, we had to consult with at least one developer from each data or API resource, in order to clarify, refine, and simplify both their and our documentation and specification details, which subsequently led to a better design of the software. For example, individual APIs return error objects in different ways, which prompted us to harmonize our error handling specification across all UBDs. In order to deliver robust software that will benefit its users and optimize software development efforts in the future, face-to-face discussions coupled with extensive Q&A email correspondence proved to be essential (Prlić and Procter, 2012). Finally, we wish to emphasize the importance that proper documentation has in a healthy software development practice (Karimzadeh and Hoffman, 2018), and its vital role in achieving our aforementioned goal.

Funding

This work was supported by ERACoSysMed Call 1 project COLOSYS (VT, JZ, MK), the COST action Gene Regulation Ensemble Effort for the Knowledge Commons [CA15205] (VT, JZ, MK, SV), the Norwegian University of Science and Technology's Strategic Research Area "NTNU Health" (VT) and the Research Council of Norway [247727/O70] (SV).

Conflict of Interest: none declared.

Acknowledgements

We would like to thank all the developers of the various data sources and web services who we consulted during the design and implementation of this work. Special thanks go to Michael Dorf and Jennifer Leigh Vendetti from BioPortal, for answering a series of long emails. We also thank EMBL-EBI software engineers Youngmi Park (EBI Search), Blake Sweeney (RNAcentral), Leonardo Gonzales (UniProt), Noemi Del Toro Ayllón (Complex Portal) and Kieron Taylor (Ensembl) for face-to-face discussions and support; and Berkeley scientist Laurent-Philippe Albou (Noctua Entity Ontology) for email feedback.

References

Howe, K. L. *et al.* (2020). Ensembl Genomes 2020—enabling non-vertebrate genomic research. *Nucleic Acids Research*, **48**(D1), D689–D695.

Jung, H. *et al.* (2009). Auto-complete for improving reliability on semantic web service framework. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 5618 LNCS, pages 36–44. Springer, Berlin, Heidelberg.

Karimzadeh, M. and Hoffman, M. M. (2018). Top considerations for creating bioinformatics software documentation. *Briefings in Bioinformatics*, **19**(4), 693–699.

Kim, J.-D. *et al.* (2019). Open Agile text mining for bioinformatics: the PubAnnotation ecosystem. *Bioinformatics*, **35**(21), 4372–4380.

Madeira, F. *et al.* (2019). The EMBL-EBI search and sequence analysis tools APIs in 2019. *Nucleic Acids Research*, **47**(W1), W636–W641.

McMurry, J. A. *et al.* (2017). Identifiers for the 21st century: How to design, provision, and reuse persistent identifiers to maximize utility and impact of life science data. *PLoS Biology*, **15**(6).

Meldal, B. H. *et al.* (2019). Complex Portal 2018: extended content and enhanced visualization tools for macromolecular complexes. *Nucleic Acids Research*, **47**(D1), D550–D558.

Prlić, A. and Procter, J. B. (2012). Ten Simple Rules for the Open Development of Scientific Software. *PLoS Computational Biology*, **8**(12), e1002802.

Sayers, E. (2010–2020). *Entrez Programming Utilities Help*. Available from: https://www.ncbi.nlm.nih.gov/books/NBK25501/.

The RNAcentral Consortium (2018). RNAcentral: a hub of information for non-coding RNA sequences. *Nucleic Acids Research*, **47**(D1), D1250–D1251.

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

Touré, V. *et al.* (2020). CausalBuilder: Bringing the MI2CAST Causal Interaction Annotation Standard to the Curator. *Preprints **2020***. (doi: 10.20944/preprints202007.0622.v1).

Vercruysse, S. and Kuiper, M. (2020). Intuitive Representation of Computable Knowledge. *Preprints **2020***. (doi: 10.20944/preprints202007.0486.v2).

Vercruysse, S. *et al.* (2020). VSM-box: General-purpose Interface for Biocuration and Knowledge Representation. *Preprints **2020***. (doi: 10.20944/preprints202007.0557.v1).

Ward, D. *et al.* (2012). Autocomplete as a research tool: A study on providing search suggestions. *Information Technology and Libraries*, **31**(4), 6–19.

Whetzel, P. L. *et al.* (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:10.20944/preprints202007.0586.v1

Zerbino, D. R. *et al.* (2018). Ensembl 2018. *Nucleic Acids Research*, **46**(D1), D754–D761.