ICBO2015
International Conference on Biomedical Ontology 2005

Proceedings of the Main Conference

July, 27-30
Lisbon, Portugal

Preface

ICBO2015 is the 6th International Conference on Biomedical Ontology held on July 26-30, 2015 in Lisboa. It is a forum for presentation and discussion of current work and new advances in Biomedical Ontologies. This conference explores directions of future research and facilitates collaboration between researchers, developers, practitioners and students.

After a careful revision, the program committee has selected 16 regular papers, 5 early career papers, 14 posters and 9 demos for presentation and discussion in the main conference. This volume contains all these papers.

The main conference includes also two invited presentations, one by Helen Parkinson, entitled *Using Ontologies in the Wild*, and another by Egon Willighagen, entitled *The role of ontologies in chem- and bioinformatics*.

The main conference was preceded by 2 workshops, 1 hackathon and 4 tutorials:

- 4th International Workshop on Vaccine and Drug Ontology Studies (VDOS)
- Third International Workshop on Definitions in Ontologies (IWOOD)
- Genomics data standards - hackathon
- Tawny-OWL: re-purposing software engineering for ontology building
- Basic Formal Ontology (BFO)
- OBO Tutorial: Best Practices for Ontology Use
- Tutorial on the Biological Pathway Exchange (BioPAX) Ontology and Pathway Commons Database

We would like to acknowledge all the organizing committee, the program committee, the additional reviewers, the volunteers, authors and attendees of ICBO, and EasyChair for providing such a useful conference management tool for free. Finally, we would like to thank all our sponsors.

July, 2015
Lisboa

Francisco Couto
Janna Hastings

Organizing Committee

Scientific Chairs: Francisco Couto (Lisboa, Portugal) and Janna Hastings (Cambridge, UK)
Local Chair: Catia Pesquita (Lisboa, Portugal)
Program Chair: Stefan Schulz (Graz, Austria)
Workshop and Tutorial Chairs: Melanie Courtot (Vancouver, Canada) and João Ferreira (Lisboa, Portugal)
Proceedings and Special Issue Chair: Dietrich Rebholz-Schuhmann (Zurich, Switzerland)
Early Career Chair: Pierre Grenon (London, UK)
Poster and Demonstrations Chair: Matthew Horridge (Stanford, USA)
Sponsorship and Publicity: Emanuel Santos (Lisboa, Portugal) and Pedro Fernandes (Oeiras, Portugal)

Program Committee

Alan Ruttenberg (University at Buffalo, USA)
Alexander D. Diehl (The Jacobs Neurological Institute, University at Buffalo, USA)
Amanda Hicks (University of Arkansas for Medical Sciences, USA)
Anika Oellrich (Sanger Institute, UK)
Barry Smith (SUNY Buffalo, USA)
Bill Hogan (University of Florida)
Christophe Lambert (Montana State University, USA)
Christopher Baker (UNB Saint John, Canada)
Colin Batchelor (Royal Society of Chemistry, UK)
Cristian Cocos (Mayo Clinic, USA)
Cui Tao (SBMI, University of Texas Health Science Center at Houston, USA)
Dagobert Soergel (Department of Library and Information Studies, University at Buffalo, USA)
Daniel Schober (Leibniz Institute of Plant Biochemistry, Germany)
Despoina Magka (Oxford University Computing Laboratory, UK)
Frank Loebe (University of Leipzig, Germany)
Fred Freitas (CIn-UFPE, Brazil)
Georgios Gkoutos (University of Cambridge, UK)
Gwen Frishkoff (Georgia State University, USA)
Harold Solbrig (Mayo Clinic, USA)
Heinrich Herre (Institute of Medical Informatics, Statistics and Epidemiology, University of Leipzig, Germany)
Helen Parkinson (European Bioinformatics Institute, UK)
James Malone (The European Bioinformatics Institute, Cambridge, UK)
James J Cimino (National Institutes of Health, USA)
James P. McCusker (5AM Solutions)
Jesualdo Tomás Fernández-Breis (Departamento de Informatica y Sistemas, Universidad de Murcia, Spain)
Jim Mccusker (5AM solutions, USA)
Jin-Dong Kim (Database Center for Life Science, Japan)
John Gennari (University of Washington, USA)
José Luís Oliveira (Universidade de Aveiro, Portugal)
Judy Blake (JAX, USA)
Laura Slaughter (Oslo University Hospital, Norway)
Lindsay Cowell (University of Texas Southwestern Medical Center at Dallas, USA)
Ludger Jansen (University of Rostock, Germany)
Mário J. Silva (Universidade de Lisboa, Portugal)
Mark Musen (Stanford University, USA)
Martin Boeker (University Medical Center Freiburg, Germany)
Mathias Brochhausen (University of Arkansas for Medical Sciences, USA)
Melissa Haendel (Oregon Health & Science University)
Michael Schroeder (TU Dresden, Germany)
Michel Dumontier (Stanford University, USA)
Nigam Shah (Stanford University, USA)
Olivier Bodenreider (US National Library of Medicine, USA)

Onard Mejino (University of Washington, USA)
Paolo Ciccarese (Harvard Medical School & Massachusetts General Hospital, USA)
Pascale Gaudet (Swiss Institute of Bioinformatics, Switzerland)
Paul Schofield (University of Cambridge, UK)
Peter Robinson (Charite Universittsmedizin Berlin, Germany)
Phillip Lord (School of Computing Science, Newcastle University, UK)
Robert Stevens (University of Manchester, UK)
Samuel Croset (European Bioinformatics Institute, UK)
Simon Jupp (European Bioinformatics Institute, UK)
Sivaram Arabandi (ONTOPRO, USA)
Suzanne E. Lewis (Lawrence Berkeley National Lab)
Werner Ceusters (SUNY at Buffalo, USA)
Yongqun He (University of Michigan, USA)

Additional Reviewers

Ahmad Bukhari, Zhe He, Matthew Horridge, Filipe Silva and Dezhao Song

Contents

Part I

Regular Papers

TNM-O an Ontology for the Tumor-Node-Metastasis Classification of Malignant Tumors: a Study on Colorectal Cancer

Martin Boeker [1,*], Fábio França [1,2], Peter Bronsert [3], and Stefan Schulz [4]

[1] : Center for Medical Biometry and Medical Informatics, University Medical Center Freiburg, Germany
[2] : Department of Informatics, University of Minho, Braga, Portugal
[3] : Center for Clinical Pathology, University Medical Center Freiburg, Germany
[4] : Institute of Medical Computer Sciences, Statistics and Documentation, Medical University of Graz, Austria

ABSTRACT

Objectives: To (1) present an ontological framework for the TNM classification system, (2) implement a TNM ontology for colon and rectum tumors based on this framework, and (3) evaluate this ontology with a classifier for pathology data.

Methods: The TNM ontology uses the Foundational Model of Anatomy for anatomical entities and BioTopLite 2 as a domain top-level ontology. General rules for the TNM system and the specific TNM classification for colorectal tumors were formulated. Additional information was collected from tumor documentation practice in an academic Comprehensive Cancer Center. Based on the ontology, an automatic classifier for pathology data was developed.

Results: TNM was represented as an information artefact which consists of single representational units. Corresponding to every representational unit, tumors and tumor aggregates were defined. Tumor aggregates consist of the primary tumor and (if existent) of infiltrated regional lymph nodes and distant metastases. TNM codes depend on the location and certain qualities of the primary tumor (T), the infiltrated regional lymph nodes (N) and the existence of distant metastases (M). Tumor data from clinical and pathological documentation were successfully classified with the ontology.

Conclusion: A first version of the TNM Ontology represents the TNM system for the description of the anatomical extent of malignant tumors. The presented work is already sufficient to show its representational correctness and completeness as well as its applicability for classification of instance data.

INTRODUCTION

Colorectal cancer is the third most common cancer worldwide and accounts for 9 % of all cancer incidence (Marmot *et al.*, 2007; Haggar and Boushey, 2009). In 2002, it affected more than one million humans. Treatment of cancer patients and research on causes of cancer are main goals of worldwide cancer control programs[1].

The premise for an evidence-based cancer treatment is a correct and unambiguous cancer diagnosis. Interdisciplinary expert groups, e.g. from clinical medicine, imaging and pathology, work closely together to establish precise tumor diagnoses (DeVita *et al.*, 2011). One of the most challenging tasks in clinical oncology is to correctly classify and code clinical findings, using a multitude of available coding systems.

Clinical and pathological staging of malignant tumors is one of the most important procedures in the diagnosis of cancer to assess prognosis and to plan the treatment necessary. The staging procedure compiles several clinical and pathological parameters: the location and the size of the *primary* tumor, the location and the number of the infiltrated *regional* lymph nodes, and the existence of distant metastases.

By far, the most important coding system for staging information is the Tumor-Node-Metastasis (TNM) classification (Sobin *et al.*, 2009) for malignant tumors, published by the Union for International Cancer Control (UICC)[2]. Despite its importance and formal precision, no logic-based representation of TNM is available so far. Such a formal representation would have several advantages over its current natural language release. An initial attempt to represent staging of lung tumors and glioma tumors was not continued (Dameron *et al.*, 2006; Marquet *et al.*, 2007).

One advantage would be the enhanced support for the TNM development and refinement. With a taxonomic backbone and axiomatic descriptions the existing complex natural language descriptions would be made explicit. This would help decompose the descriptions into all their defining criteria. This could help to detect errors, inconsistencies and ambiguities in definitions (Ceusters *et al.*, 2004; Cornet and Abu-Hanna, 2005). Many combinations of tumor findings are difficult to code due to ambiguos or overlapping criteria (non-disjoint definitions) or non-exhaustive definitions, which often results in cases where no TNM code or more than one TNM code is applicable to a given tumor state.

Additionally, logical inconsistencies and coding problems due to complexity could be detected earlier by automated reasoning. Description logics (DL) would here be the method of choice. Such a TNM DL ontology could be further used for automatic classification of instance data from clinical databases on a sound logical basis. Advanced retrieval and querying tools would be additional benefits. For these use cases, a formalized TNM version could constitute a unified source from which a variety of clinical documentation and analysis tools could be derived.

With this work we propose to close the gap of a missing formal representation by outlining and prototyping a TNM ontology (TNM-O).

Following up initial attempts in the breast cancer domain (Boeker *et al.*, 2014) the objectives of this work are (1) to present an ontological

*to whom correspondence should be addressed

[1] http://www.who.int/cancer/modules/en/

[2] http://www.uicc.org

framework for the TNM classification system, (2) to implement a TNM ontology describing colon and rectum tumors based on this framework, and (3) to evaluate this ontology with a classifier for pathology data.

The TNM classification

The UICC published the first edition of the TNM coding system based on the anatomic extent of disease (EOD) in 1968. Since then, the system has undergone several revisions and arrived in 2009 at the 7th edition. The objectives of the TNM classification are six-fold. It supports treatment planning, prediction of outcomes (prognosis), evaluation of treatment results, exchange of information between different participants in the treatment process, continuing research in malignant diseases, and cancer control (Sobin *et al.*, 2009; Webber *et al.*, 2014).

The TNM coding procedure requires a high degree of both domain knowledge and experience in tumor documentation. Even documentation experts frequently engage in discussions about how a given case should be coded correctly. This is mainly due to the development of the TNM classification as an evolutionary process (Webber *et al.*, 2014), which has to account for the huge amount of new scientific insights in tumor prognosis and the dependency of therapeutic effects on tumor stage. Controlled by medical experts, TNM's underlying structure has become more and more complex over the years. Experts in different fields of oncology require for a change in TNM maintenance representing the increasing complexity, the separation from clinical practice and the resources needed for documentation (Quirke *et al.*, 2010, 2007).

Dependent on the location of the primary tumor, the three parts of the code (T, N, and M) represent different aspects of a tumor. *T* describes size and sometimes infiltrative level of the *primary* tumor, *N* describes infiltrated regional lymph nodes, and *M* distant metastases. T and N usually provide three to four levels with increasing severity, *viz.* T0–T3 and N0–N3, respectively. For distant metastases, there is only a binary classification into M1 (evidence) an M0 (no evidence).

The results from the *clinical* assessment have to be accurately discerned from the *pathological* assessment due to their different meanings and evidence levels. This distinction is symbolized by a prefix *c* (clinical) and *p* (pathological) for most primary tumor locations.

Many users of the TNM struggle with the correct coding as well as with the interpretation of TNM codes. This is one of the reasons for the need in improvement of tumor documentation and coding in primary documentation, clinical studies and cancer registries (Abernethy *et al.*, 2009; Aumann *et al.*, 2012; Nagtegaal *et al.*, 2000).The classification of the different primary tumor locations differs to the same extent as the underlying diseases. As a consequence, even expert coders resp. physicians in one organ system might encounter difficulties in the correct application or interpretation of TNM to a different organ system.

Besides the complex semantics of the main numeral TNM codes, a series of additional symbols exists, which might have largely different meanings in the different tumor locations. Prefixes, suffixes, and certainty factors increase the confusion, e.g. for *carcinoma in situ* the suffix "is" has to be used (Tis). With the possibility to always use a code of "X" if the underlying clinical or pathological situation provides incomplete information, inaccurate and incomplete code assignments become widespread (MX for "no statement on metastases possible").

Type	ICO-O 3 morphology
adenocarcinoma	8140/3
Mucinous adenocarcinoma	8480/3
Signet-ring cell carcinoma	8490/3
Small cell carcinoma	8041/3
Squamous cell carcinoma	8070/3
Adenosquamous carcinoma	8560/3
Medullary carcinoma	8510/3
Undifferentiated carcinoma	8020/3

Table 1. ICD-O 3 morphology codes for tumors of the colon and the rectum

METHODS

The TNM ontology uses the Foundational Model of Anatomy (Rosse and Mejino Jr., 2003) for anatomical entities and BioTopLite 2 as a domain top-level ontology (Beißwanger *et al.*, 2008; Schulz and Boeker, 2013). Tailored for the biomedical domain and based on description logics (Baader *et al.*, 2007), BioTopLite 2 (BTL2) provides upper-level types both for general categories like *Material object*, *Process*, *Information object*, *Quality* etc., as well as constraints on all of them, using a set of sixteen canonical relations, partly derived from the OBO Relation Ontology (RO) (Smith *et al.*, 2005). They constrain each category by means of a set of general class axioms. It also contains other axioms such as relationship chains, existential and value restrictions. Thus, the building of domain ontologies under BTL2 heavily constrains the freedom of the ontology engineer, which is fully intended as this guarantees a higher predictability of the domain ontologies produced under BTL2.

The general rules for the TNM system and the specific TNM for tumors of the colon and the rectum (ICD-O topography chapters C18-C21, for ICD-O morphology codes see Table 1) were represented as described in Sobin et al. (2009) and Hamilton et al. (2000).

A classifier for individuals (instances) derived from pathology reports was developed employing the OWL API (version 4.0.1)[3] and the HermIT DL reasoner (version 1.3.8)[4]. It classifies breast tumor and colorectal tumor data based on the corresponding TNM ontologies. The classifier reads either tabular input data from files or can process data from manual entry via a graphical user interface.

RESULTS

TNM-O is designed as a modular system of independent ontologies. For every organ or organ system based module of the TNM classification system, TNM-O provides a specific set of ontologies. The TNM connecting ontology serves as a hub to import BioTopLite2 as well as the organ and organ system specific TNM ontologies (see Table 2). The modular architecture allows for inclusion of only those modules which are actually needed by an application.

[3] http://owlapi.sourceforge.net/

[4] http://hermit-reasoner.com/

Without inclusion of BioTopLite2, the TNM hub ontology has the description logic expressivity of ALC. It consists of 79 axioms, 38 logical axioms, and 39 classes. It includes 35 subClassOf and 1 EquivalentTo axioms. Most of the classes are proxy classes to BioTopLite2. Inclusion of BioTopLite2 changes the DL expressivity to SRI.

The TNM ontology for colorectal tumors has the description logic expressivity of ALC. For TNM version 7.0 (version 6.0 in brackets), it consists of 386 (357) axioms, 291 (199) logical axioms, and 158 (149) classes. It includes 177 (160) subClassOf, 21 (18) EquivalentTo and 18 (18) DisjointClasses axioms.

Representational units in the TNM-Ontology

The representation of the TNM system is decomposed in representational units T, N and M and the location of the primary tumor. Thus, for every existing code Tn, Nn and Mn in combination with a specific organ there exists one *TNM-O:RepresentationalUnit* which is an *btl2:InformationObject*. E.g. every TNM code for colorectal cancer is represented by a separate class. These classes are connected with their patho-anatomical relata of type *PrimaryTumor* or *TumorAggregate* by the relation **btl2:isRepresentedBy**. In the remaining text, the namespace of the TNM ontology is suppressed for clarity:

TumorOfColonAndRectumWith7OrMoreMetastaticRegionalLymphNodes
subClassOf
TumorAggregate and
btl2:isRepresentedBy some
(*ColonRectumTNM_pN2b* or *ColonRectumTNM_N2b*) and
btl2:isRepresentedBy only
(*ColonRectumTNM_pN2b* or *ColonRectumTNM_N2b*)

Representation of the primary tumor

The primary tumor is represented as *PrimaryTumor*, a subclass of *MalignantAnatomicalStructure*. The characteristics relevant for the representational unit *T* of the TNM classification system are represented as location and qualities of *PrimaryTumor*. For colorectal tumors the exact localization of the tumor in the gut wall, the quality of the tumor confinement with respect to neighboring organs (confined or invasive), the quality of the assessment (no assessment, no evidence or carcinoma in situ), are important. *PrimaryTumor* is directly related to the corresponding representational unit:

InvasiveTumorOfSubmocosaOfColonAndRectum EquivalentTo
ColonAndRectumTumor and
(**btl2:isBearerOf** some (*Confinement* and
(**btl2:projectsOnto** some *Invasive*))) and
(**btl2:isIncludedIn** some
SubmocosaOfLargeIntestine)

InvasiveTumorOfSubmocosaOfColonAndRectum subClassOf
btl2:isRepresentedBy some
(*ColonRectumTNM_T1* or
ColonRectumTNM_pT1) and
btl2:isRepresentedBy only
(*ColonRectumTNM_T1* or
ColonRectumTNM_pT1)

Name	Description
BioTopLite2	Upper domain level ontology
TNM-O	TNM-O central connecting ontology
TNM-O_breast_7	TNM-O for breast cancer (TNM version 7) in: Boeker et al. (2014)
TNM-O_colorectal_6	TNM-O for colorectal cancer (TNM version 6)
TNM-O_colorectal_7	TNM-O for colorectal cancer (TNM version 7)

Table 2. Modular structure of TNM-O. Codes in clinical documentation and cancer registries are versioned by TNM versions. The meaning of codes and stages changes between versions. The modular structure is designed to include versions for every available TNM encoded entity (tumor location) so that the intended meaning is preserved according to the version which was used for coding.

Representation of regional lymph nodes

The most complex part of the TNM classification of many primary tumor locations is the interpretation of the axis *N*, which describes to which extent the primary tumor has infiltrated regional lymph nodes. The anatomy of lymph nodes draining the colon and rectum was modeled according to clinical anatomical conventions. Metastatic regional lymph nodes can exactly be located by the exact subclass of infiltrated regional lymph node:

MetastaticLymphNodeOfColonAndRectumTumor EquivalentTo
LymphNode and
(**btl2:hasPart** some
MetastasisOfColonAndRectumTumor)

MetastaticRegionalLymphNodeOfColonAndRectumTumor EquivalentTo
MetastaticLymphNodeOfColonAndRectumTumor and
ColonAndRectumRegionalLymphNode

To define regional lymph node metastases of colorectal cancers, the aggregate of primary tumor and infiltrated lymph nodes around the colon and rectum (*TumorAggregate*) has to be considered as one (composite) entity. The representational unit *N* of the TNM classification of colorectal cancers is dependent on the count of metastatic regional lymph nodes and the presence of subserosal tumor deposits without regional lymph node metastases. The count of metastatic lymph nodes is represented by subclasses of *CardinalityValueRegion*:

TumorOfColonAndRectumWith2or3MetastaticRegionalLymphNodes
EquivalentTo
TumorOfColonAndRectumWith1to3MetastaticRegionalLymphNodes and
(**btl2:isBearerOf** some
(*Cardinality* and
(**btl2:projectsOnto** some
Cardinality2or3) and
(**btl2:projectsOnto** only
Cardinality2or3)))

Representation of distant metastases

For the representational unit *M* of the TNM classification system the existence and number of distant metastases is evaluated. The definition of distant metastases excludes *regional* lymph nodes as their localization:

DistantMetastasisOfColonAndRectumTumor EquivalentTo
MetastasisOfColonAndRectumTumor and

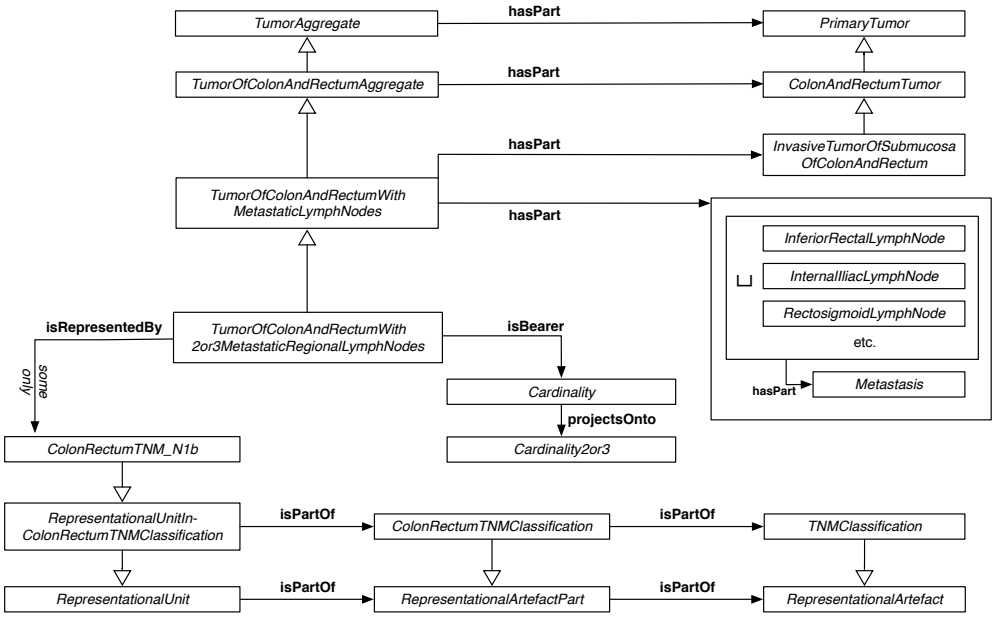


Abbildung 1: Graph of the patho-anatomical structures represented by an N1b representational unit of the TNM-O for colorectal tumors version 7 (TNM-O_colorectal_7.owl). T and M representational units are unspecified.

(not (**btl2:isIncludedIn** some *ColonAndRectumRegionalLymphNode*)) and (**btl2:isIncludedIn** some *BodyPart*)

TumorOfColonAndRectumWithDistantMetastasis EquivalentTo *TumorOfColonAndRectumAggregate* and (**btl2:hasPart** some *DistantMetastasisOfColonAndRectumTumor*)

TumorOfMammaryGlandWithDistantMetastasis subClassOf (**btl2:isRepresentedBy** only (*MammaryGlandTNM_M1* or *MammaryGlandTNM_pM1*)

The hub TNM Ontology for all tumors can be downloaded from http://purl.org/tnmo/TNM-O.owl. The ontologies for breast tumors and colorectal tumors are named according to Table 2 and can be downloaded from the same site. They need to be loaded in the hub ontology.

Classification of pathology data

We classified data on the state of regional lymph nodes (TNM: N) of 382 specimens of colorectal carcinomas which were documented at the Institute of Surgical Pathology, Medical Center – University of Freiburg. All data were coded as RDF-OWL instance data and classified by an application based on the OWL API using an OWL classifier[5]. Automatic classification was solely based on axioms defined in the colorectal TNM-O version 7 (TNM-O_colon_7.owl). Criteria employed from instance data are shown in table 3.

All instance data could be classified to classes of the ontology. *A-posteriori* comparison of the classification results with the findings from the pathology database by an experienced pathologist showed 100 % correct classification results.

Criterium	btl2 superclass	Value
primary tumor extension	MaterialObject	Epithelium, Submucosa, Lamina propria, Subserosa, Adventitia, VisceralPeritoneum
primary tumor growth pattern	Quality	Infiltrative, Confined
primary tumor epistemology	Quality	NoAssessment NoEvidence
regional LN number	Quality	Cardinality1 Cardinality2or3 Cardinality4to6 Cardinality7orMore
regional LN epistemology	Quality	NoAssessment NoEvidence
distant Mx location	MaterialObject	Peritoneum
distant Mx no. of organs	Quality	Cardinality1 Cardinality2orMore
distant Mx epistemology	Quality	NoEvidence

Table 3. Criteria of TNM version 7 for colorectal cancers. All TNM codes can be inferred from this criteria. The exact wording of the textual definitions of the TNM in version 7 is diverging[6]. Exact count of infiltrated organs in distant metastasis is omitted.

[5] http://owlapi.sourceforge.net/

[6] http://cancerstaging.blogspot.de/2005/02/colon-and-rectum.html

DISCUSSION

TNM is a globally accepted system to describe the anatomical extent of malignant tumors (Sobin *et al.*, 2009; Webber *et al.*, 2014). Although TNM is of high importance for the staging of tumor diseases, to the knowledge of the authors, there exists no formal representation of TNM so far. With this work, the authors provide a first outline of a TNM ontology (TNM-O) and a prototypical implementation of TNM for colorectal cancers. This work also shows also that TNM-O classifies instance data.

Over time, TNM has developed into a coding system which had to accommodate both the pragmatics of coding and representational accuracy. The literature on ambiguities and difficulties of TNM in practice is abundant. The discussion of TNM for breast tumors illustrates the dilemma of its maintainers (Barr and Baum, 1992; Gusterson, 2003; Güth *et al.*, 2007). They had to account for the rapid progression of scientific knowledge on tumors and to keep it usable at the same time: new versions of TNM were already outdated when compared with new scientific insights. On the other hand, it became increasingly complex, with a negative impact on usability by non-expert and expert documentation staff and physicians.

This study is limited as far we provide here a *first version* of the TNM Ontology (TNM-O) which has been developed only for mammary gland (Boeker *et al.*, 2014) and colorectal tumors. As these two tumor entities are the most complex and best represented ones in TNM, the durrent version is already as far complete and stable to be used as a blueprint for TNM-O extensions to other organ systems.

Due to the nature of the domain and the rich top-level ontology employed, the computational resources needed to classify the ontology are considerable. To alleviate performance issues TNM-O will be provided as modules for different organ systems. Thus, the users can import only the modules of interest into their application context.

Future research should evaluate the presented prototype ontology (i) by implementing further tumor locations, and (ii) by systematical application in clinical classification and retrieval scenarios. We will provide the formalization of TNM for other primary tumor locations in a modular way, so that users can select which part of the TNM-O they would like to use. In this way, we hope to reduce the computational resources already needed to a minimum.

Conclusion

We presented a first version of an ontology (TNM-O) that represents the TNM tumor classification system. The presented work is already sufficient to show the representational correctness and completeness of the TNM-O as well as its applicability for classification of instance data. This work provides a foundation for an exhaustive TNM ontology.

REFERENCES

Abernethy, A. P., Herndon, J. E., Wheeler, J. L., Rowe, K., Marcello, J., and Patwardhan, M. (2009) Poor Documentation Prevents Adequate Assessment of Quality Metrics in Colorectal Cancer. *JOP*, **5**, 167–174.

Aumann, K., Amann, D., Gumpp, V., Hauschke, D., Kayser, G., May, A. M., Wetterauer, U., and Werner, M. (2012) Template-based synoptic reports improve the quality of pathology reports of prostatectomy specimens. *Histopathology*, **60**, 634–644.

Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., and Patel-Schneider, P. F. (2007) The Description Logic Handbook: Theory, Implementation, and Applications, 2nd Edition 2nd ed. Cambridge University Press.

Barr, L. C. and Baum, M. (1992) Time to abandon TNM staging of breast cancer? *The Lancet*, **339**, 915–917.

Beißwanger, E., Schulz, S., Stenzhorn, H., and Hahn, U. (2008) BioTop: An Upper Domain Ontology for the Life Sciences - A Description of its Current Structure, Contents, and Interfaces to OBO Ontologies. *Applied Ontology*, **3**, 205–212.

Boeker, M., Faria, R., and Schulz, S. (2014) A Proposal for an Ontology for the Tumor-Node-Metastasis Classification of Malignant Tumors: a Study on Breast Tumors. In, Jansen,L. (eds) *et al.*, *Ontologies and Data in Life Sciences (ODLS 2014)*, IMISE-REPORTS. Leipzig.

Ceusters, W., Smith, B., Kumar, A., and Dhaen, C. (2004) Ontology-based error detection in SNOMED-CT (R). In, Fieschi,M. (eds) *et al.*, *Medinfo 2004: Proceedings of the 11th World Congress on Medical Informatics, Pt 1 and 2*. IOS Press, Amsterdam, pp. 482–486.

Cornet, R. and Abu-Hanna, A. (2005) Description logic-based methods for auditing frame-based medical terminological systems. *Artificial Intelligence in Medicine*, **34**, 201–217.

Dameron, O., Roques, É., Rubin, D., Marquet, G., and Burgun, A. (2006) Grading lung tumors using OWL-DL based reasoning. In, *Presentation Abstracts*. Stanford, USA, p. 69.

DeVita, V. T., Lawrence, T. S., and Rosenberg, S. A. eds. (2011) DeVita, Hellman, and Rosenberg's cancer: principles & practice of oncology 9th ed. Wolters Kluwer Health/Lippincott Williams & Wilkins, Philadelphia.

Gusterson, B. A. (2003) The new TNM classification and micrometastases. *The Breast*, **12**, 387–390.

Güth, U., Jane Huang, D., Holzgreve, W., Wight, E., and Singer, G. (2007) T4 breast cancer under closer inspection: A case for revision of the TNM classification. *The Breast*, **16**, 625–636.

Haggar, F. A. and Boushey, R. P. (2009) Colorectal Cancer Epidemiology: Incidence, Mortality, Survival, and Risk Factors. *Clin Colon Rectal Surg*, **22**, 191–197.

Hamilton, S. R., Aaltonen, L. A., Cancer, I. A. for R. on, Organization, W. H., and others (2000) Pathology and genetics of tumours of the digestive system IARC press Lyon:

Marmot, M., Atinmo, T., Byers, T., Chen, J., Hirohata, T., Jackson, A., James, W., Kolonel, L., Kumanyika, S., Leitzmann, C., Mann, J., Powers, H., Reddy, K., Riboli, E., Rivera, J. A., and Schatzkin, A.*et al.* (2007) Food, Nutrition, Physical Activity, and the Prevention of Cancer: a Global Perspective World Cancer Research Fund / American Institute for Cancer Research, Washington DC, US.

Marquet, G., Dameron, O., Saikali, S., Mosser, J., and Burgun, A. (2007) Grading glioma tumors using OWL-DL and NCI Thesaurus. *AMIA Annu Symp Proc*, **2007**, 508–512.

Nagtegaal, I. D., Kranenbarg, E. K., Hermans, J., Velde, C. J. H. van de, Krieken, J. H. J. M. van, and Committee, the P. R. (2000) Pathology Data in the Central Databases of Multicenter Randomized Trials Need to Be Based on Pathology Reports and Controlled by Trained Quality Managers. *JCO*, **18**, 1771–1779.

Quirke, P., Cuvelier, C., Ensari, A., Glimelius, B., Laurberg, S., Ortiz, H., Piard, F., Punt, C. J., Glenthoj, A., Pennickx, F., Seymour, M., Valentini, V., Williams, G., and Nagtegaal, I. D. (2010) Evidence-based medicine: the time has come to set standards for staging. *J. Pathol.*, **221**, 357–360.

Quirke, P., Williams, G. T., Ectors, N., Ensari, A., Piard, F., and Nagtegaal, I. (2007) The future of the TNM staging system in colorectal cancer: time for a debate? *The Lancet Oncology*, **8**, 651–657.

Rosse, C. and Mejino Jr., J. L. (2003) A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *Journal of Biomedical Informatics*, **36**, 478–500.

Schulz, S. and Boeker, M. (2013) BioTopLite: An Upper Level Ontology for the Life Sciences. Evolution, Design and Application. In, Hornbach,M. (ed), *INFORMATIK 2013. Ontologien in den Lebenswissenschaften.*, Lecture Notes in Informatics. Gesellschaft für Informatik, Bonn, pp. 1889–1899.

Smith, B., Ceusters, W., Klagges, B., Köhler, J., Kumar, A., Lomax, J., Mungall, C., Neuhaus, F., Rector, A. L., and Rosse, C. (2005) Relations in biomedical ontologies. *Genome Biology*, **6**, R46.

Sobin, L. H., Gospodarowicz, M. K., and Wittekind, C. (2009) TNM Classification of Malignant Tumours 7. John Wiley & Sons, Chichester, West Sussex, UK ; Hoboken, NJ.

Webber, C., Gospodarowicz, M., Sobin, L. H., Wittekind, C., Greene, F. L., Mason, M. D., Compton, C., Brierley, J., and Groome, P. A. (2014) Improving the TNM classification: Findings from a 10-year continuous literature review. *Int. J. Cancer*, **135**, 371–378.

An ontological analysis of diagnostic assertions in electronic healthcare records

Werner Ceusters[1,*] and William R. Hogan[2]
[1] Department of Biomedical Informatics, University at Buffalo, 921 Main street, Buffalo, USA
[2] Department of Health Outcomes and Policy, University of Florida, 2004 Mowry Rd, Gainesville, Florida, USA

ABSTRACT

We present a comparative analysis of two sets of Referent Tracking Tuples (RTT), which each author of this paper crafted independently from the other and which are about the same portion of reality that one could assume to be described faithfully through registered diagnoses in the problem list of an electronic healthcare record system. The analysis thereby focused on (1) the choice of particulars that each of the authors deemed necessary and sufficient for an accurate description, (2) what these particulars are instances of, (3) how they relate to each other, and (4) the motivations of each author for the choices made. It was found that despite the large variety in RTTs crafted, there was wide, though not total, agreement about the appropriateness of the choices made. Disagreements arose from various issues such as potential lack of orthogonality in the OBO Foundry and in some cases on what types the classes in the ontologies represent. The authors' main source of disagreement was due to different interpretations of the literature on Information Content Entities (ICEs).

1 INTRODUCTION

When during a clinical encounter a provider establishes a diagnosis for a patient under his care, he typically enters one or more diagnostic codes into that patient's electronic healthcare record (EHR). When the same patient is later seen by a distinct provider, for instance for a second opinion or for a reason not related to the first encounter, this second provider will also enter one or more diagnostic codes. Patients' records tend to accumulate many of these diagnostic assertions, specifically when the providers are working in the same EHR, or when such information is transferred from one to the other. They are also accumulated when records from various systems are merged into data warehouses equipped with master patient index facilities. There is a large variety amongst EHR systems and data warehouse interfaces in how they display such diagnostic information, a small hypothetical example being shown in Table 1.

A problem with information provided in this way, is that it is not possible to construct a completely accurate view on what is (and has been) the case in reality (Rector et al, 1991). A question which, in relation to the information in Table 1, cannot be answered reliably is whether the two diagnoses are about the very same disorder the patient suffers from (thereby highlighting different aspects of that disorder which cannot be expressed using a single ICD-code) or about two distinct disorders the patient suffers from simultaneously. Other questions are, for example, whether the persons that *entered* the diagnoses also *made* the diagnoses, how the dates when diagnoses were entered relate to the dates when the diagnoses were actually made, and so forth.

1	Patient ID	Diagnosis		Date entered	Entered by
2	1234	274.9:	Gout, unspecified	9-1-2014	J. Doe
3	1234	715.97:	Osteoarthrosis, unspecified whether generalized or localized, ankle and foot	9-1-2014	S. Thump

Table 1: Two diagnoses provided on the same day, about the same patient, entered by two distinct EHR users.

Referent Tracking (RT) is a methodology to avoid, resolve, and document these sorts of ambiguities in EHRs (Ceusters & Smith, 2006). This is achieved by building data stores composed of Referent Tracking Tuples (RTT). The core part of an RTT expresses a relationship that obtains between a particular—globally and singularly uniquely identified in the realm of the RT System used to generate (and track usage of) Instance Unique Identifiers (IUIs)—and either another particular or a universal (or defined class), representations of which are – ideally – taken from one or more ontologies that follow the principles of Ontological Realism (Ceusters & Manzoor, 2010; Smith & Ceusters, 2010). Whenever a continuant is referenced in an RTT, time indexing is used following the conventions outlined in (Smith et al., 2005). As an example, the following RTT—formulated in simplified abstract syntax—asserts that there exists a particular to whom IUI '#4' is assigned, and that this particular is an instance of human being during the time period to which the IUI 't5' is assigned:

#4 ***instance-of*** HUMAN BEING ***at*** t5 (Ex.1)

The methodology was expanded in (Ceusters et al., 2014) to translate datasets into assertions such that not only the portion of reality (POR) described by the dataset and the dataset itself are represented, but so also the relations between components of this dataset on the one hand and the corresponding PORs on the other hand.

The purpose of the work reported here was to assess to what extent the authors of this paper—two experts in RT—would be able to develop independently from one another a collection of RTTs that describe the same POR in a semantically-interoperable way. The analysis presented is the first

* To whom correspondence should be addressed: ceusters@buffalo.edu

step in this endeavor and focuses on (1) the choice of particulars deemed necessary and sufficient for an accurate description of the selected POR, (2) what these particulars are instances of, (3) how they relate to each other, and (4) the motivations of each author for the choices made.

2 METHODS

The POR selected for the experiment was the one ambiguously described in Table 1. Since the goal of the exercise was not to identify nor, when possible, resolve ambiguities, it was further specified that the diagnoses were about the same disorder, in the sense as formulated in the foundations for the Ontology of General Medical Science (OGMS) (Scheuermann et al., 2009). No instructions were given on what ontologies to use, or in what format to provide the RTTs. Results were exchanged in a password-protected file and the passwords disclosed after each author acknowledged receipt of the other's result. The authors then compared the original RTTs in stepwise fashion. The first step was to identify the particulars that both authors referred to in their assertions. Since both authors assigned IUIs independently, thereby assigning distinct IUIs to the very same particulars, a second step was then to re-assign IUIs as if the collection of RTTs was merged into one single RT system, thereby still keeping track of which RTT was asserted by which author. In a third step, this collection was then analyzed and differences in representations discussed, however without paying attention to the temporal indexing required for RTTs describing a POR in which a continuant is involved.

3 RESULTS

Table 2 lists the particulars and what they are instances of as originally—thus prior to comparison of the proposed representations—argued for by the author hereafter referred to as 'X'. Table 3 does so for author Y. Each row represents part of an RTT asserting that the particular denoted in the 'IUI'-column is an instance of the universal denoted by the representational unit (RU) in the 'Class'-column, drawn from the ontology named in the 'Ontology'-column. The description relates the particulars informally to the scenario analyzed. The column labeled 'Ind.' contains the IUIs of the Information Content Entities (ICE) of which the RTTs themselves are concretizations. The columns 'Y' and 'X' contain scores reflecting how Y, resp. X, after discussion considered the RTT appropriate, '0' meaning 'not at all', '1' 'ok, but', and '2' 'absolutely'. Table 4 and 5 list for X and Y respectively the RTTs involving non-instantiation relationships. An IUI or Index in bold indicates that the corresponding POR is referred to by both authors. Author X listed 21 particulars involving 23 instantiations; Y did so for 28 particulars involving 1 instantiation each, not counting in both cases as particulars the temporal regions related to the time-indexing required for certain RTTs.

Ind.	IUI	Description	Ontology	Class	Y	X
T1	**P1**	the patient	OBI	Homo sapiens	1	2
T2	**P2**	the doctor who made diagnosis #1	OBI	Homo sapiens	1	2
T3	**P3**	the doctor who made diagnosis #2	OBI	Homo sapiens	1	2
T4	**P4**	diagnosis #1	OGMS	Diagnosis	2	2
T5	**P5**	diagnosis #2	OGMS	Diagnosis	2	2
T6	P6	the disorder the patient has	OGMS	Disorder	2	2
T7	P6		DO	Gout	1	1
T8	P6		DO	Osteoarthritis	1	1
T9	P7	entry in problem list for diagnosis #1		Dataset record	2	2
T10	P8	entry in problem list for diagnosis #2		Dataset record	2	2
T11	**P9**	the process of doctor #1 making diagnosis #1	OGMS	Diagnostic process	2	2
T12	**P10**	the process of doctor #2 making diagnosis #2	OGMS	Diagnostic process	2	2
T13	P11	doctor #1's doctor role	OMRSE	Physician role	2	2
T14	P12	doctor #2's doctor role	OMRSE	Physician role	2	2
T15	P13	the patient's patient role	OMRSE	Patient role	2	2
T16	P14	EHR		EHR	2	2
T17	**P15**	patient ID cell of entry #1		Denotator	1	1
T18	**P16**	diagnosis cell of entry #1		Denotator	1	1
T19	P17	doctor cell of entry #1		Denotator	1	2
T20	**P18**	diagnosis cell of entry #2		Denotator	1	1
T21	P19	date cell of entry #2		Denotator	1	2
T22	P20	ICD-9-CM coding system			2	2
T23	P21	patient's afflicted foot	FMA	FMA:Foot	2	2

Table 2: Particulars and what they are instances of from the original perspective of author 'X'.

39 distinct particulars were identified, 10 of them by both authors. For only 2 (RTTs T2 and T3) did both authors select the same instantiating universal while for 2 others (T4 and T5) universals were selected from distinct ontologies, but with a close, nevertheless debatable, match. For the remaining 6, the universals chosen stand in is-a relations.

X drew 9 classes from 5 realism-based ontologies – the OGMS, the Ontology of Medically Related Social Entities (OMRSE), the Foundational Model of Anatomy (FMA), the Disease Ontology (DO) and the Ontology of Biomedical Investigations (OBI)—and identified the need for three more classes—'denotator', 'EHR' and 'dataset record', for which no realism-based ontology was found. Y used 9 classes drawn from 4 realism-based ontologies, 2 of which (FMA and OGMS) were also used by X, and 2 distinct ones: the Basic Formal Ontology (BFO) and the Information Artifact Ontology (IAO). He also identified the need for 2 classes currently without an ontological home: 'patient identifier' and 'ICD-9-CM code and label', as well as 2 classes (Gout and Osteoarthrosis, R49 and R51 in Table 5) for which he did not identify any particular as being required for an accurate description of the scenario.

53 particular-to-particular relationships in total were represented: 22 alone by X, 27 alone by Y and 4 (R14, R21, R22 and R23 in Table 4 and 5) by both authors, be it nevertheless through distinct, yet synonymous formulations. Y listed also two RTTs, each one expressing aboutness between a particular and a universal (R49 and R51, Table 5).

Ind.	IUI	Description	Ontology	Class	Y	X
T24	P22	the ICE which is concretized in the spreadsheet you might be looking at	IAO	ICE	2	2
T25	P23	the portion of chalk on the blackboard which make up what we call 'that spreadsheet'	BFO	Material entity	2	2
T26	P24	the pattern of chalk lines, spaces, characters, etc., in that portion of chalk	BFO	Quality	2	2
T27	**P1**	the material entity whose ID is '1234' in the spreadsheet	BFO	Material entity	1	1
T28	**P15**	the patient identifier which is concretized in each first cell of the 2nd and 3rd row of the concretization of P22		Patient identifier	2	2
T29	P25	the portion of chalk making up the text string '1234' in the first cell of the 2nd row	BFO	Material entity	2	2
T30	P26	the quality in P25 which makes P25 a concretization bearer	BFO	Quality	2	2
T31	P27	portion of chalk making up the string '1234' in the 1st cell of the 3rd row of the spreadsheet	BFO	Material entity	2	2
T32	P28	the quality in P27 which makes P27 a concretization bearer	BFO	Quality	2	2
T4	**P4**	the diagnosis which is concretized in the first two cells of the 2nd row of the concretization of P22 in front of your eyes	OGMS	Diagnosis	2	2
T33	P29	the quality through which **P4** is concretized	BFO	Quality	2	2
T5	**P5**	the diagnosis concretized in the first two cells of the 3rd row of the concretization of P22 in front of your eyes	OGMS	Diagnosis	2	2
T34	P30	the quality through which **P5** is concretized	BFO	Quality	2	2
T2	**P2**	the person whose name is 'J. Doe' in the spreadsheet	FMA	Human being	1	1
T3	**P3**	the person whose name is 'S. Thump' in the spreadsheet	FMA	Human being	1	1
T35	P31	the clinical picture about **P1** available to **P2** and **P3**	OGMS	Clinical picture	2	2
T36	P32	part of the life of **P1** which is described in P31	OGMS	Bodily process	1	1
T37	**P9**	the interpretive process which resulted in **P4**	OGMS	Bodily process	1	2
T38	**P10**	the interpretive process which resulted in **P5**	OGMS	Bodily process	2	2
T39	P33	the disease in **P1**	OGMS	Disease	2	2
T40	**P16**	the ICE concretized in the 2nd cell of the 2nd row		Icd-9-cm code and label		
T41	P34	the quality through which **P16** is concretized	BFO	Quality	2	2
T42	**P18**	the ICE concretized in the 2nd cell of the 3rd row		Icd-9-cm code and label	2	2
T43	P35	the quality through which **P18** is concretized	BFO	Quality	2	2
T44	P36	the process of, as we say 'entering' diagnosis 1 in the EHR'	BFO	Process	2	2
T45	P37	the quality of some part of some hard disk which concretizes d1	BFO	Quality	2	2
T46	P38	the process of, as we say 'entering' diagnosis 2 in the EHR'	BFO	Process	2	2
T47	P39	the quality of some part of some hard disk which concretizes diagnosis 2	BFO	Quality	2	2

Table 3: Particulars and what they are instances of from the perspective of author 'Y'.

Ind.	:	RTT in abstract syntax without time-component			Y	X
R1	:	**P1**	RO:bearer of	P13	2	2
R2	:	**P1**	RO:has part	P6	2	2
R3	:	**P1**	RO:has part	P21	2	2
R4	:	**P10**	RO:realizes	P12	2	2
R5	:	T7	corresponds with	**P16**	2	2
R6	:	T8	corresponds with	**P18**	2	2
R7	:	**P15**	RO:part of	P7	1	1
R8	:	**P15**	RO:part of	P8	1	1
R9	:	**P15**	IAO:denotes	**P1**	0	1
R10	:	**P16**	RO:part of	P7	1	1
R11	:	**P16**	IAO:denotes	**P4**	1	1
R12	:	P17	RO:part of	P7	1	1
R13	:	P17	IAO:denotes	**P2**	0	1
R14	:	**P2**	RO:agent of	**P9**	2	2
R15	:	**P2**	RO:bearer of	P11	2	2
R16	:	**P18**	RO:part of	P8	1	1
R17	:	**P18**	IAO:denotes	**P5**	0	1
R18	:	P19	IAO:denotes	**P3**	0	1
R19	:	P21	RO:has part	P6	1	1
R20	:	**P3**	RO:bearer of	P12	2	2
R21	:	**P3**	RO:agent of	**P10**	2	2
R22	:	**P4**	OBI:is specified output of	**P9**	2	2
R23	:	**P5**	OBI:is specified output of	**P10**	2	2
R24	:	P7	IAO:is about	P6	2	2
R25	:	P8	IAO:is about	P6	2	2
R26	:	P9	RO:realizes	P11	2	2

Table 4: particular to particular relationships listed by author X

Ind.	:	RTT in abstract syntax without time component			Y	X
R27	:	P24	inheres-in	P23	2	2
R28	:	P24	concretizes	P22	2	2
R29	:	**P15**	part-of	P22	2	2
R30	:	P25	bears-concretization-of	**P15**	2	2
R31	:	P26	inheres-in	P25	2	2
R32	:	P26	is-about	**P1**	2	2
R33	:	P27	bears-concretization-of	**P15**	2	2
R34	:	P28	inheres-in	P27	2	2
R35	:	P28	is-about	**P1**	2	2
R36	:	P29	concretizes	**P4**	2	2
R37	:	P29	is-about	**P1**	2	2
R38	:	P29	is-about	P33	2	2
R39	:	P30	concretizes	**P5**	2	2
R40	:	P30	is-about	**P1**	2	2
R41	:	P30	is-about	P33	2	2
R42	:	**P2**	agent-of	P36	2	2
R43	:	**P3**	agent-of	P38	2	2
R44	:	P32	has-participant	**P1**	2	2
R22	:	**P9**	creates	**P4**	2	2
R14	:	**P9**	has-agent	**P2**	2	2
R45	:	**P9**	has-input	P31	2	2
R23	:	**P10**	creates	**P5**	2	2
R21	:	**P10**	has-agent	**P3**	2	2
R46	:	**P10**	has-input	P31	2	2
R47	:	P33	inheres-in	**P1**	2	2
R48	:	P34	concretizes	**P16**	2	2
R49	:	P34	is-about	GOUT	2	2
R50	:	P35	concretizes	**P18**	2	2
R51	:	P35	is-about	OSTEOARTHROSIS	2	2
R52	:	P36	creates	P37	2	2
R53	:	P37	concretizes	**P4**	2	2
R54	:	P38	creates	P39	2	2
R55	:	P39	concretizes	**P5**	2	2

Table 5: relationships other than instance-of listed by author Y.

X indicated from which ontologies the relationships were drawn. Y used relations from the BFO 2.0 Draft Specifications, or under discussion in the context of the IAO.

4 DISCUSSION

Despite the large variation in RTTs crafted for what at first sight looks like a simple POR, there was after discussion wide, though not total agreement, about the appropriateness of the choices made (agreement is indicated by the same scores appearing in the X and Y columns of Tables 2-5). 'Appropriateness' is here to be measured in terms of what an optimal collection of RTTs for the POR under scrutiny would be since one could argue that the ground truth for what is expressed in EHR entries is largely unknown. The 'ground truth' is thus much broader than just what the patient had (this being part of the non-assertional part of the POR): it includes what the clinicians stated about what the patient had (these statements being part of the assertional part of the POR). If what the patient precisely had cannot be inferred from what was stated, it would be wrong to construct a collection of RTTs that states that the patient has such or such a specific type of disorder. To represent the non-assertional part of a POR that a collection of assertions is about, one has to resort to these assertions and to what has already been established to be the case through other means. The optimal collection of RTTs would be the one which satisfies the following criteria: (1) it consists of RTTs which describe the non-assertional part of the POR only to the extent to which there is enough evidence for what those RTTs themselves assert to be true (e.g. there is sufficient evidence that the patients are human beings, there is not sufficient evidence that the diagnoses are correct) and (2) it consists of other RTTs which describe the assertional part in relation to the RTTs referenced under (1).

We note here that the level of disagreement in the representations of X and Y do not invalidate the RT method, but rather reflect the need for uniform conventions on which ontologies and relations to use, as well as problems in the ontological theories, their implementations, and documentation that were available to represent the scenario. We return to this issue as we discuss the major sources of disagreement. Indeed, as will become clear, this work shows that the RT method is a stringent test of ontologies.

Although both authors agreed on the necessary existence of the patient (P1) and the two clinicians (P2, P3) for the analyzed scenario to be faithful to reality, they each selected distinct universals to assert instantiation. X represented P1 as a human with a patient role. Y represented P1 as a material entity without assigning a patient role, his choice of material entity being motivated by the fact that P1 has been a material entity all the time through its existence, but not a human (e.g., it was a zygote at a time prior to being human). This difference in representation is related to the temporal indexing that RT requires for continuants, an element not further discussed in this paper. But given the two authors' temporal indexing, both authors agree that each other's instantiations were correct.

Both authors disagreed though about how to interpret the representational units for the universal *Human being* from the selected ontologies. Y used 'human being' as synonym for the FMA's 'human body' class, although FMA does not list synonyms. Y argued against X's 'Homo sapiens' taken from OBI based on its linking to other ontologies in Ontobee, which altogether seem to confuse 'Homo sapiens' as an instance of the universal 'species' with those instances of organism that belong to – but are not instances of – the species 'Homo sapiens'. X counters that despite the use of species names, 'Homo sapiens' and similar classes in OBI all descend from a class called 'organism'. Also, the 'Homo sapiens' class in OBI has synonyms 'Human being' and 'human'. It would be an enormous task indeed to find non-taxonomic names for every type of organism in the world and refactor ontologies based on the NCBI Taxonomy on this basis. The problem here is the lack of face value of terms selected as class names in the respective ontologies.

Both authors agreed on the existence of a disorder and a disease resulting from it in the patient, as well as two diagnoses and the two distinct processes that generated each. Both authors also agree that none of these entities should be confused or conflated: nothing at the same time can be an instance of two or more of the following: disease, disorder, diagnosis, and diagnostic process.

A problem is that the Disease Ontology selected by X, confuses not only disorders and disease, but also disease courses. For instance 'physical disorder' in DO is a subtype of 'disease', in direct contradiction to OGMS. X agrees that DO is flawed, however it is the only ontology of disease that at least purports to strive for compliance with realist principles, and represents an improvement over flawed medical terminologies such as SNOMED-CT and NCI Thesaurus. If perfection were a requirement to use an ontology, we could make no progress. Nevertheless, the persistent, glaring flaws of DO from the perspective of OGMS give serious pause on using it accurately and precisely.

Both authors offered a different perspective on what parts of Table 1 actually constitute a diagnosis. They agreed that any such table—whether presented as a problem list on a video monitor or tablet as the scenario worked with by X, or as a spreadsheet drawn with chalk on a blackboard as envisioned by Y—is built out of continuants that are concretizations of instances of ICE reflecting a diagnosis. But whereas X identified the mere concretization of the ICD-code and label to be denoting the diagnosis, Y argued that also the concretization of the patient identifier is part of that which concretizes the diagnosis (a requirement for the diagnosis to stand in an 'is about' relation to the patient). This, and a large amount of other differences in representation, were due to distinct interpretations of the literature on the nitty-gritty of how to deal with ICE and concretizations thereof, how instances of ICE relate to other instances of ICE, and what exactly the relata are of relationships such as aboutness

and denotation. Whereas, for instance, X committed to ICE being parts of other ICE, Y commits only to parthood of the independent continuants in which inhere the qualities that concretize the corresponding ICE, without making that clear in the representation however. Another key issue with ICE is that Y represented the qualities concretizing the ICE as being about something, whereas X followed the IAO where the ICE itself denotes or is-about something. After further analysis both authors agreed that Y's representation is better and that it advances the theory of ICE in IAO.

Although it was a priori agreed upon that the patient in the scenario would have only one disorder, an ambiguity that was left open was whether both diagnoses were actually correct: so Table 1 could be interpreted in distinct ways: (1) both diagnoses are correct from a medical perspective and describe distinct aspects of the same disease, or (2) at least one diagnosis is wrong. Also, because RT tuples contain provenance as to whom is the source of the statement contained therein – note that Ex.1 above is a simplified representation not containing the provenance information – X interpreted both (1) the RT tuple that instantiated the disease as gout (by Doe) and (2) the RT tuple that instantiated it as osteoarthritis (by Thump) as being faithful representations of what Thump and Doe believed at the time they formulated their diagnoses. X did not believe himself to be recognizing both diagnoses as straightforwardly accurate and therefore resorted in his representation to a mechanism offered in RT to craft RTTs about RTTs that are later found to have been based on a misunderstanding of the reality at the time they were crafted (Ceusters, 2007). Y crafted a representation that does not commit to what specific disease type(s) the patient's disease actually is an instance of. This was achieved by representing the diagnoses to be simultaneously about the patient on the one hand (in contrast to X who represents the diagnoses to be about the disorder/ disease itself), and about the disease universals – gout and osteoarthrosis resp. – denoted by the respective ICD-codes and labels on the other hand. This aboutness-relation between an instance of ICE and a universal can be represented in RT but of course cannot be represented in OWL without recourse to workarounds such as those discussed by Schulz et al (2014).

Although both authors resorted to OGMS for a large part of their RTTs, differences in representation were observed because of the source material consulted: X used the OGMS OWL artifact as basis, whereas Y used the definitions and descriptions in the paper that led to the development of OGMS (Scheuermann et al., 2009).

5 CONCLUSION

In representing a common scenario in healthcare which EHR data are about, the two authors agreed on key entities including the patient, doctors, diagnoses, and the processes by which the doctors generated the diagnosis. Although they agreed in general about the types instantiated by the particulars in the scenario, and how the particulars are related to each other, they chose different representational units and relations from different ontologies due to various issues such as potential lack of orthogonality in the OBO Foundry and in some cases disagreement on what types the classes in the ontologies represent. These distinctions exist, not because the authors entertained distinct competing conceptualizations, but because they expressed matters differently.

Differences in the choice of ontologies constitute a risk: distinct ontologies may represent reality from distinct perspectives and despite being veridical might not be derivable from each other because the axioms required to do so would be missing, for the simple reason that such axioms would fall outside the purpose of the specific ontologies. This would lead the representations by each of the authors not to be semantically interoperable unless additional ontology bridging axioms would be crafted. The authors' main source of disagreement was due to different interpretations of the literature on ICEs, which ultimately led to a planned reformulation of the theory of ICE and reference. Although this study is limited by the participation of only 2 subjects and the analysis of one report, it highlights the fact that the RT method and the clarity and precision it requires in representing reality is a powerful tool in identifying areas of needed improvement in existing, realism-based ontologies.

ACKNOWLEDGEMENTS

This work was supported by award UL1 TR000064 from the National Center for Advancing Translational Sciences. This paper is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

REFERENCES

Ceusters, W. (2007). Dealing with mistakes in a referent tracking system. In H. KS (Ed.), *Proceedings of ontology for the intelligence community 2007 (oic-2007)* (pp. 5-8). Columbia MA.

Ceusters, W., Chiun Yu Hsu, & Smith, B. (2014). Clinical data wrangling using ontological realism and referent tracking. *CEUR Workshop Proceedings, 1237*, 27-32.

Ceusters, W., & Manzoor, S. (2010). How to track absolutely everything? In L. Obrst, T. Janssen & W. Ceusters (Eds.), *Ontologies and semantic technologies for the intelligence community. Frontiers in artificial intelligence and applications.* (pp. 13-36). Amsterdam: IOS Press.

Ceusters, W., Smith, B. (2006). Strategies for referent tracking in electronic health records. *Journal Biomedical Informatics, 39*(3), 362-378.

Rector, A.L., Nowlan, W.A., & Kay, S. (1991). Foundations for an electronic medical record. *Methods Inf Med, 30*(3), 179-186.

Scheuermann, R.H., Ceusters, W., & Smith, B. (2009). Toward an ontological treatment of disease and diagnosis. *Summit on Translat Bioinforma, 2009*, 116-120.

Schulz, S., Martínez-Costa, C., Karlsson, D., Cornet, R., Brochhausen, M., Rector, A. (2014). An ontological analysis of reference in health record statements. In: *Proceedings of FOIS 2014*, 2014.

Smith, B., & Ceusters, W. (2010). Ontological realism: A methodology for coordinated evolution of scientific ontologies. *Applied Ontology, 5*(3-4), 139-188. doi: 10.3233/Ao-2010-0079

Smith, B., Ceusters, W., . . . Rosse, C. (2005). Relations in biomedical ontologies. *Genome Biology, 6*(5): R46. doi:10.1186/Gb-2005-6-5-R46

A UML Profile for Functional Modeling Applied to the Molecular Function Ontology

Patryk Burek [1]*, Frank Loebe [2] and Heinrich Herre [1]

[1]Institute for Medical Informatics, Statistics and Epidemiology (IMISE), University of Leipzig, Haertelstrasse 16-18, 04107 Leipzig, Germany

[2]Computer Science Institute, University of Leipzig, Augustusplatz 10, 04109 Leipzig, Germany

ABSTRACT

Gene Ontology (GO) is the largest, and steadily growing, resource for cataloging gene products. Naturally, its growth raises issues regarding its structure. Modeling and refactoring big ontologies such as GO is far from being simple. It seems that human-friendly graphical modeling languages, such as the Unified Modeling Language (UML) could be helpful for that task. In the current paper we investigate if UML can be utilized for making the structural organization of the Molecular Function Ontology (MFO), a sub-ontology of GO, more explicit. In addition, we examine if and how using UML can support the refactoring of MFO. We utilize UML and its extension mechanism for the definition of a UML dialect, which is suited for modeling functions and is called Function Modeling Language (FuML). Next, we use FuML for capturing the structure of molecular functions. Finally, we propose and demonstrate some refactoring options for MFO.

1 INTRODUCTION

The Molecular Function Ontology (MFO) is a sub-ontology of the Gene Ontology (GO) – the largest, and steadily growing, resource for cataloging gene products. In 2000 GO contained less than 5,000 terms, in 2003 – 13,000 (Gene Ontology Consortium, 2004), in 2010 it exceeded 30,000 (du Plessis *et al.*, 2011), whereas at the beginning of 2015 its size is above 42,000 terms. The growth of the ontology leads to a suboptimal structure (du Plessis *et al.*, 2011), which motivates refactoring initiatives such as (Guardia *et al.*, 2012; Alterovitz *et al.*, 2010), besides the work of the GO Consortium itself that constantly improves and evolves GO. It turns out that modeling and refactoring big ontologies such as GO is a difficult task, the realization of which can be supported by a human-friendly representation format. The serialization formats used for machine processing of the ontologies, such as the OBO flat file format (Horrocks, 2007) or the Web Ontology Language (OWL) (W3C OWL Working Group, 2009), are not the easiest for a human user. This motivates the adoption of human-friendly graphical notations like those used in software engineering for the task of ontology representation (Kogut *et al.*, 2002; Belghiat and Bourahla, 2012) for certain purposes.

The de facto standard for graphical conceptual modeling of software systems is the Unified Modeling Language (UML) (Rumbaugh *et al.*, 2005), currently developed and maintained by the Object Management Group (OMG) (Object Management Group, 2014). UML has a big potential for various applications that go beyond software engineering, among them for modeling biological knowledge and biological ontologies (Shegogue and Zheng, 2005; Guardia *et al.*, 2012).

UML is well-suited for modeling biological systems, not at least due to the rich infrastructure and the available tools. In particular, the UML built-in extension mechanisms such as stereotypes and profiles permit the easy construction of domain- or task-specific UML dialects, e.g the OBO relations profile (Guardia *et al.*, 2012). Numerous tools for UML modeling are available on the market and can be used out of the box for visualizing biological ontologies as a whole or in part.

In the present paper we investigate if UML can be utilized for making the structure of MFO more explicit and if it can support the refactoring of MFO. We use UML and its extension mechanism for the definition of a UML dialect, called Function Modeling Language (FuML), which is suited for function modeling. Next, we use FuML for modeling the structure of molecular functions. Finally, we propose and demonstrate some refactoring options for MFO.

2 METHODS

2.1 Molecular Function Ontology

Like all GO terms, functions in MFO are specified by id, name, natural language definition and an optional list of synonyms. For instance, the function of catalyzing carbohydrate transmembrane transport is specified by id: GO:0015144; name: *carbohydrate transmembrane transporter activity*; definition: catalysis of the transfer of carbohydrate from one side of the membrane to the other; synonym: sugar transporter. Additionally, for each function its relations with other concepts can be captured. The semantics of the relations that are used for this purpose is provided by serialization languages such as the OBO flat file format or OWL, and/or by the OBO relations ontology (RO) (Smith *et al.*, 2005). In particular, functions in MFO are organized into a hierarchy by means of the is_a link from RO; furthermore, they are linked with processes by the part_of relationship from RO; and in some cases they have relations with concepts of other ontologies such as ChEBI (Degtyarenko *et al.*, 2008). For instance, GO:0015144 is linked, by means of the RO is_a relation, to its parent functions GO:1901476 *carbohydrate transporter activity* and GO:0022891 *substrate-specific transmembrane transporter activity*, by means of the RO part_of relation to the process GO:0034219: *carbohydrate transmembrane transport*, and by means of the RO transports_or_maintains_localization_of to CHEBI:16646: *carbohydrate*.

From the above we see that the semantics of functions in MFO is provided to a large extent by informal natural language expressions and partially by relations with other concepts.

*To whom correspondence should be addressed: patryk.burek@imise.uni-leipzig.de

2.2 Intensional Subsumption

We propose defining the notion of function subsumption, which is a backbone of MFO, upon an intensional interpretation of the is_a relation. Typically, in the field of ontology engineering the extensional aspect of the is_a relation is stressed; in OWL, for instance, A is a subclass of B if every instance of A is an instance of B. The same interpretation is used in RO, where is_a is defined by the reference to the sets of all instances (extensions) of the concepts. According to this understanding the is_a relation is often called extensional subsumption, in contrast to its intensional counterpart(s), where we focus on structural subsumption (Woods, 1991). Instead of referring to instances, this type of subsumption is defined based on the structure of concepts. The latter can be understood as a composition of conceptual parts by means of various composing relations. For illustration within GO itself, GO:0005215: *transporter activity* is justified to intensionally subsume GO:0022857: *transmembrane transporter activity*, because, following (Woods, 1991), both are activities and they are (partially) defined by part-of relations, to GO:0006810: *transport* and to GO:0055085: *transmembrane transport*, resp., and the latter is subsumed by the former. Overall, the main assumption is that concepts are complex structures which can be organized into a subsumption hierarchy. The reading of intensional subsumption is similar to inheritance in object-oriented languages, where one class inherits its structure from another. That enables the structuring of classes into hierarchies.

2.3 UML Profiles and FuML

UML is a graphical modeling language founded on the explicit distinction between the static and the dynamic views of a system; it introduces thirteen diagram types, grouped into two sets: structural modeling diagrams and behavioral modeling diagrams. UML lacks constructs dedicated to function modeling as such, but it provides several build-in mechanisms that allow for an easy extension of the language. Among them are profiles.

A profile is a light-weight UML mechanism, typically used for extending the language for particular platforms, domains or tasks. It specifies a set of extensions of the UML standard metamodel which include, among others, stereotypes. With stereotypes it is possible to extend the standard UML vocabulary with new model elements. A stereotype can be graphically represented by a dedicated icon, though in the most straightforward form it is represented by a stereotype name, surrounded by guillemets and placed above the name of the stereotyped UML element, cf. «Function» in Figure 1.

We used the profile mechanism for developing a UML extension, called Function Modeling Language (FuML), aimed at supporting the modeling of functions, function ascription, and function decomposition. FuML defines 15 stereotypes for representing functions and function structure, 8 stereotypes for modeling function decomposition, subsumption and function dependencies. The full specification of FuML stereotypes is provided in (Burek and Herre, 2014). In the remaining part of the current paper we analyze how far FuML can be used for modeling and refactoring MFO.

3 ANALYSIS

3.1 Modeling Molecular Functions with FuML

3.1.1 Functions FuML enables graphical modeling of functions in a compact and in an extended form. The compact form is particularly suited for big models containing many functions, whereas the extended form is designed for visualizing the dependencies within the structure of a single function or between several functions.

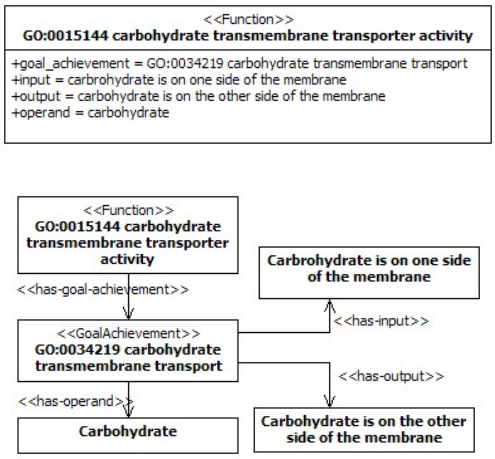


Figure 1. A FuML model of a molecular function, displayed in the compact notation at the top and in the extended form at the bottom.

Figure 1 presents an exemplary FuML model, depicting the structure of the function GO:0015144: *carbohydrate transmembrane transporter activity*. The upper part of the figure presents the compact notation, whereas the extended notation is shown in the lower part. The stereotypes utilized in the figure are discussed in the remainder of the current section.

A function in FuML is interpreted as a role that an entity plays in the context of some goal achievement, such as e.g. a teleological process. This account of functions is similar to (Karp, 2000), where a biological function of a molecule is described as the role that the molecule plays in a biological process. In this sense, the function GO:0015144: *carbohydrate transmembrane transporter activity*, defined in GO as "catalysis of the transfer of carbohydrate from one side of the membrane to the other", depicts the catalyst role in the teleological process of transferring carbohydrate from one side of the membrane to the other. In terms of the structure we can therefore say that a function specification contains as its part a specification of a goal achievement, understood as a teleological entity which is specified in terms of a transformation from an input situation to an output situation. As presented in Figure 1, a function is depicted by a UML classifier with a stereotype «Function». It connects to its goal achievement by an association with a stereotype «has-goal-achievement» in the extended notation, whereas the compact notation utilizes the attribute goal_achievement.

3.1.2 Goal Achievements In FuML, a goal achievement (GA) x is defined as a category the instances of which are transitions from certain input situations to output situations. Input and output are defined as follows:

- The input category y of the goal achievement x is a situation category such that every instance of x is a transition starting from a situation instantiating y.

- The output y of a goal achievement x is a situation category specifying the situations in which instances of x result by transition. Every instance of x is a transition resulting in a situation instantiating y.

For example, the goal achievement *carbohydrate transmembrane transport* establishes the input category, the instances of which are situations of carbohydrate being on the one side of the membrane, and the output category, the instances of which are situations of carbohydrate being on the other side of the membrane. This means that every instance of *carbohydrate transmembrane transport* exhibits a transition from an instance of the input category to an instance of the output category, i.e. from individual situations of carbohydrate located on one side of the membrane, to individual situations of carbohydrate located on the other side of the membrane.

As shown in Figure 1, an input is indicated in the extended notation by the association with stereotype «has-input», and by the input attribute of a function in the compact notation. The representation of outputs is analogous.

Typically, a transformation from an input to an output situation is a process, and then the GA can be understood as a process category. In the running example, the GA is a teleological process category, namely of carbohydrate transfer from one side of the membrane to the other. This process exhibits the causal transition from the situation of carbohydrate being on one side of the membrane to the situation where carbohydrate is on the other side of the membrane.

3.1.3 Mode of Goal Achievement In some cases the specification of a function is not reduced to a mere input-output pair, but it defines constraints on the method of function realization. For example, the molecular functions GO:0015399: *primary active transmembrane transporter activity* and GO:0015291: *secondary active transmembrane transporter activity* share the same input: solute is on one side of the membrane, and the same output: solute is on the other side of the membrane. Therefore, the pure input-output views of the functions are equal. However, they are distinct due to the way in which they achieve the goal. The former function is realized by means of some primary energy source, for instance, a chemical, electrical or solar source, whereas the latter relies on a uniporter, symporter or antiporter protein. Thus we see that the functions provide the same answer to the question on *what* is to be achieved, however they provide different answers on *how* that is realized. In order to represent this distinction, in FuML we introduce another component of function structure, called *Mode of Goal Achievement*. The mode x of the goal achievement y specifies the way in which y transforms the input to the output situation. For GO:0015399 the mode is: some primary energy source, for instance chemical, electrical or solar source, and for GO:0015291 it is: uniporter, symporter or antiporter protein. The mode is a constraint on the function realization, which does not affect the input or the output. For example, if one adds to the function of transmembrane transport the constraint that the transport should be realized by the uniporter protein then the input and the output remain unchanged. However, the function as such changes in that not every transportation process realizes it, but only those that are driven by a uniporter protein.

3.1.4 Participants Often goal achievements are expressed by action sentences of natural language and thus the results of linguistic analysis of action sentences can be applied to the analysis of the structure of goal achievements. In linguistics, the role that a noun phrase plays with respect to the action or state described by the verb of a sentence is called a thematic role (Harley, 2010). The specifications of molecular functions in MFO often contain two thematic roles – a patient (called an operand in FuML) and an actor (called a doer in FuML). An operand indicates the entity undergoing the effect of the action. We say that an operand y of the goal achievement x specifies a category y such that instances of x operate on instances of y. GO:0015144 operates on (transports) carbohydrate.

A doer is not as common in MFO as an operand. For example, in the discussed carbohydrate transmembrane transport function no doer is indicated. Typically, a doer is a part of the GA in cases where the mode of realization is provided. For instance, the functions GO:0015292 *uniporter activity* and GO:0015293 *symporter activity* both specify the mode of realization and each indicates its doer, namely the respective protein.

4 PATTERNS OF FUNCTION SUBSUMPTION

Behind functional subsumption actually various distinct relations are implicitly hidden (Burek *et al.*, 2009). FuML introduces several distinct patterns for function subsumption (Burek and Herre, 2014). In the following section we discuss the application of three of those patterns for the modeling of MFO.

In FuML the notion of function subsumption is founded on the subsumption of goal achievements. We say that the function x is subsumed by the function y if the goal achievement of x is subsumed by the goal achievement of y. Since goal achievements are quite complex entities, it is not trivial to answer the question of what it means that one goal achievement subsumes another. Here, however, the analysis of GA structure is helpful, which pertains to the intensional aspects of the corresponding GA category, as discussed in previous sections. Based on this approach one can detect various patterns of function subsumption.

4.1 Operand Specialization

Since function specifications often contain operands, it is very common to construct a hierarchy of functions on the basis of the taxonomic hierarchy of their operands. In fact, this pattern is applied frequently in MFO. Consider, for instance, the functions GO:0015075: *ion transmembrane transporter activity* and GO:0008324: *cation transmembrane transporter activity*, linked by the is_a relation in GO. The relation between those two functions is based on the relation of their operands, as cation is subsumed by ion. In FuML function subsumption by operand specialization is depicted with a dependency link with stereotype «operand-spec». The supplier of the link is the subsumed function and the client is the subsumer.

4.2 Mode Addition

Another pattern of function subsumption, frequently met in MFO, is based on modes of goal achievement. Consider two functions, GO:0022857: *transmembrane transporter activity* and GO:0022804: *active transmembrane transporter activity*. Both share the same operand, namely substance, as well as the same input-output pair – operand is on one side of the membrane and operand is on the other side of the membrane. In this sense those functions are equal. However, they differ in that the former does not define any mode of realization, whereas the latter has the following mode defined: the transporter binding the solute undergoes a series of conformational changes. Therefore, one can say that

GO:0022804 specializes GO:0022857 by addition of a mode. We say that function x is subsumed by the function y by mode addition if x is subsumed by y and x has some mode, whereas y has no mode assigned. Function subsumption by mode addition is depicted in FuML by means of a dependency link with stereotype «mode-added». The subsumed function is the supplier of the link and the subsuming function is a client.

4.3 Mode Specialization

Subsumption of functions can be based on the mode of realization also in cases where a parent function has already a mode assigned. Consider, for instance, the function GO:0022804: *active transmembrane transporter activity* having the mode: transporter binds the solute and undergoes a series of conformational changes and the function GO:0015291: *secondary active transmembrane transporter activity* with the mode: transporter binds the solute and undergoes a series of conformational changes driven by chemiosmotic energy sources, including uniport, symport or antiport. The latter clearly characterizes particular modes of active transmembrane transport. Consequently, it seems intuitive to say that GO:0015291 specializes GO:0022804 (as is the case in GO). We call this type of function subsumption the subsumption by mode specialization and define it as follows: The function x is subsumed by the function y by mode specialization if x is subsumed by y and mode r of x specializes mode s of y. In FuML function subsumption by mode specialization is depicted with a dependency link with stereotype «mode-spec». The subsumed function is the supplier of the link and the specialized function is a client.

5 APPLICATION

The application of FuML to GO pursues two objectives. The first objective is the usage of FuML for establishing a semantic basis for molecular functions that supports the representation of functions in an organized way beyond the textual description. Moreover, the discussed patterns represent basic knowledge on the interrelations between biological processes and molecular functions. The part_of relation between biological processes and molecular functions can be mapped to the has-goal-achievement association between functions and goal achievements.

The second and the main objective of applying FuML to MFO is to explicitly document design choices and the subsumption patterns utilized implicitly in MFO. Figure 2 presents such a documentation for a fragment of MFO in terms of FuML. The patterns are indicated by stereotypes of FuML, which enables an easy-to-grasp visualization of the structure of MFO as well as the underlying design choices. One benefit of this approach is that the explicit specification of the design choices makes the ontology much more intelligible for a human user.

Furthermore, the application of FuML reveals potential of refactoring and revision of GO. For instance, the application of FuML in modeling the functions GO:0022857: *transmembrane transporter activity* and GO:0022891: *substrate-specific transmembrane transporter activity* reveals that both share similar goal achievements: transfer of an operand from one side of a membrane to the other, with input: operand is on one side of the membrane, and output: operand is on the other side of the membrane. Consequently and following FuML, a potential difference between GO:0022857 and GO:0022891 can be searched in their operands. For GO:0022857 that is a substance, whereas for GO:0022891 it is a specific substance or group of substances. Therefore, the first refactoring option would be to explicitly document the pattern of subsumption between GO:0022857 and GO:0022891 as operand specialization. The alternative refactoring option is driven by the further analysis of operands of those functions, in particular by clarifying what the difference between "a substance" and "a specific substance or group of substances" is. The answer could be found in GO:0022892 *substrate-specific transporter activity*, a parent function of GO:0022891. An operand of GO:0022892 is exemplified by macromolecules, small molecules or ions. In that case, however, it seems that functions like GO:0090482: *vitamin transmembrane transporter activity* and GO:0015238: *drug transmembrane transporter activity* should also be considered as substance specific transmembrane transport and specialize GO:0022891 by operand specialization, which is currently not the case, however.

Finally, the third possible refactoring option could be based on the assumption that the distinction between those two operands is only superficial and GO:0022891 is merely used for the organization of the function taxonomy, i.e., for grouping all functions that are distinguished by operands such as ion, alcohol, and water. According to this view, GO:0022891 would in fact be a duplication of GO:0022857, introduced into MFO only for the purpose of structuring it, but not as a specification of particular biological functions. As illustrated in Figure 2, FuML enables the replacement of GO:0022891 with an explicit specification of the design choices by stereotyped links.

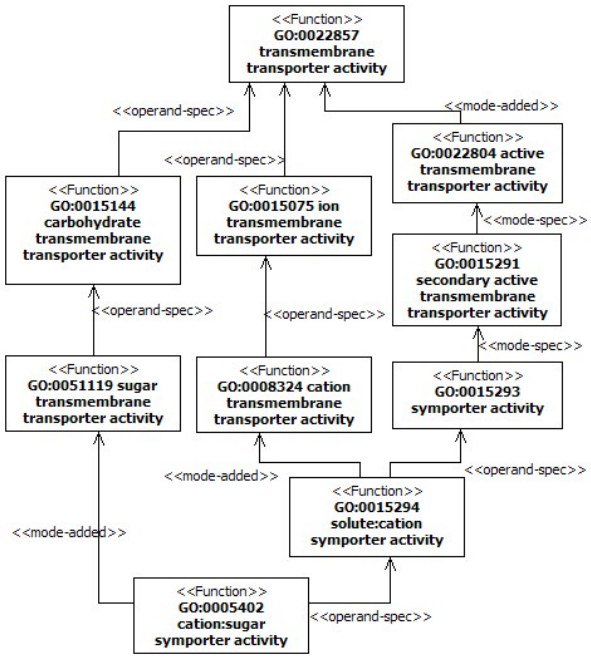


Figure 2. An MFO segment modeled with FuML.

The decision on the refactoring option, as in any modeling enterprise, is the responsibility of the modeler(s), GO developers in this case. Yet, the above analysis demonstrates how graphical languages,

such as FuML, similarly as in software and systems engineering, can drive and support that task for biological ontologies such as MFO.

6 RELATED WORK

The ideas underlying the structure of functions, introduced in FuML, are the result of an analysis of the current state of the art of function modeling in software, systems and ontological engineering. For instance, the interpretation of a function in terms of a role is common not only in biological systems (Karp, 2000), but also in functional modeling in mechanical engineering (Kitamura *et al.*, 2006; Lind, 1994; Chandrasekaran and Josephson, 2000).

The notion of goal achievement grasps the teleological character of a function, its orientation on some goal. This aspect is stressed in many approaches to function representation, e.g. (Sasajima *et al.*, 1995; Iwasaki *et al.*, 1995; Gero, 1990). In particular, defining a function in terms of input-output pairs is present in modeling technical artifacts (Borgo *et al.*, 2011; Goel *et al.*, 2009).

The mode of realization, also called the way-of-function-achievement, specifying the constraints on the method of function realization is present in (Kitamura *et al.*, 2002), among others.

To the best of our knowledge, the presented patterns of function decomposition are not collected and integrated into any other single modeling framework, though the techniques themselves are commonly used, especially in software and systems engineering, e.g. see the function-means-context link in (Bracewell and Wallace, 2001) or the decomposition with zig-zaging in (Nam, 2001).

7 CONCLUSION

In the current paper we present and discuss applications of UML and patterns for function subsumption to the modeling and refactoring of biological ontologies. In particular, we developed a UML profile for functional modeling, called the Function Modeling Language (FuML) (Burek and Herre, 2014), and apply it to the modeling and refactoring of a segment of the Molecular Function Ontology.

The application of FuML enables the systematic, graphical representation of information that is currently available in MFO mainly in the form of textual descriptions. We demonstrate that behind the extensional is_a relation, which is used for the construction of MFO, several different patterns of intensional subsumption can be determined. Modeling MFO via FuML helps in identifying these patterns and, moreover, provides the means for representing them directly in the hierarchy of molecular functions. We argue that this can help making the ontology structure more comprehensible for human users and supports communication. The claim is illustrated by an analysis and a model of an MFO fragment with FuML, from which we derive several refactoring options.

Besides proposing FuML and the particular refactoring options in this paper, for future work we consider first the continued analysis of MFO. Extending this to a larger scale may require establishing software support, e.g., for identifying subsumption pattern instances within MFO (semi-)automatically. Moreover, FuML and its methods may also be transferred to or yield new methods for common languages of biomedical ontologies, nowadays including OWL.

REFERENCES

Alterovitz, G., Xiang, M., Hill, D. P., Lomax, J., Liu, J., Cherkassky, M., Dreyfuss, J., Mungall, C., Harris, M. A., Dolan, M. E., *et al.* (2010). Ontology engineering. *Nature biotechnology*, **28**(2), 128–130.

Belghiat, A. and Bourahla, M. (2012). Automatic generation of OWL ontologies from UML class diagrams based on meta-modelling and graph grammars. *World Academy of Science, Engineering and Technology*, **6**(8), 380–385.

Borgo, S., Carrara, M., Garbacz, P., and Vermaas, P. E. (2011). A formalization of functions as operations on flows. *Journal of Computing and Information Science in Engineering*, **11**(3), 031007.

Bracewell, R. H. and Wallace, K. M. (2001). Designing a representation to support function-means based synthesis of mechanical design solutions. In S. Culley, A. Duffy, C. McMahon, and K. Wallace, editors, *Proceedings of ICED01, Glasgow, Scotland, UK, Aug 21-23*, pages 275–282.

Burek, P. and Herre, H. (2014). FuML Specification v1.0. Onto-med report, University of Leipzig.

Burek, P., Herre, H., and Loebe, F. (2009). Ontological analysis of functional decomposition. In H. Fujita and V. Mařík, editors, *Proceedings of the 8th International Conference on Software Methodologies, Tools and Techniques, SoMeT 2009, Prague, Czech Republic, Sep 23-25*, pages 428–439, Amsterdam. IOS Press.

Chandrasekaran, B. and Josephson, J. R. (2000). Function in device representation. *Engineering with computers*, **16**(3-4), 162–177.

Degtyarenko, K., De Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcántara, R., Darsow, M., Guedj, M., and Ashburner, M. (2008). ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic acids research*, **36**(Suppl 1), D344–D350.

du Plessis, L., Škunca, N., and Dessimoz, C. (2011). The what, where, how and why of gene ontology — a primer for bioinformaticians. *Briefings in Bioinformatics*, **12**(6), 723–735.

Gene Ontology Consortium (2004). The Gene Ontology (GO) database and informatics resource. *Nucleic acids research*, **32**(Suppl 1), D258–D261.

Gero, J. S. (1990). Design prototypes: a knowledge representation schema for design. *AI Magazine*, **11**(4), 26–36.

Goel, A. K., Rugaber, S., and Vattam, S. (2009). Structure, behavior, and function of complex systems: The structure, behavior, and function modeling language. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, **23**(01), 23–35.

Guardia, G. D., Vêncio, R. Z., and de Farias, C. R. (2012). A UML profile for the OBO relation ontology. *BMC Genomics*, **13**(Suppl 5), S3.

Harley, H. (2010). Thematic roles. *The Cambridge Encyclopedia of the Language Sciences*, pages 861–862.

Horrocks, I. (2007). OBO flat file format syntax and semantics and mapping to OWL Web Ontology Language. Technical report, University of Manchester.

Iwasaki, Y., Vescovi, M., Fikes, R., and Chandrasekaran, B. (1995). Causal functional representation language with behavior-based semantics. *Applied Artificial Intelligence: An International Journal*, **9**(1), 5–31.

Karp, P. D. (2000). An ontology for biological function based on molecular interactions. *Bioinformatics*, **16**(3), 269–285.

Kitamura, Y., Sano, T., Namba, K., and Mizoguchi, R. (2002). A functional concept ontology and its application to automatic identification of functional structures. *Advanced Engineering Informatics*, **16**(2), 145–163.

Kitamura, Y., Koji, Y., and Mizoguchi, R. (2006). An ontological model of device function: industrial deployment and lessons learned. *Applied Ontology*, **1**(3), 237–262.

Kogut, P., Cranefield, S., Hart, L., Dutra, M., Baclawski, K., Kokar, M., and Smith, J. (2002). UML for ontology development. *The Knowledge Engineering Review*, **17**(1), 61–64.

Lind, M. (1994). Modeling goals and functions of complex industrial plants. *Applied Artificial Intelligence: An International Journal*, **8**(2), 259–283.

Nam, P. S. (2001). *Axiomatic design: Advances and applications*. Oxford University Press, New York.

Object Management Group (2014). `http://www.omg.org/`.

Rumbaugh, J., Jacobson, I., and Booch, G. (2005). *The Unified Modeling Language Reference Manual*. Addison Wesley, Reading, Massachusetts, 2. edition.

Sasajima, M., Kitamura, Y., Ikeda, M., and Mizoguchi, R. (1995). FBRL: A function and behavior representation language. In *Proc. of IJCAI 1995*, pages 1830–1836.

Shegogue, D. and Zheng, W. J. (2005). Integration of the Gene Ontology into an object-oriented architecture. *BMC Bioinformatics*, **6**(1), 113.

Smith, B., Ceusters, W., Klagges, B., Köhler, J., Kumar, A., Lomax, J., Mungall, C., Neuhaus, F., Rector, A. L., and Rosse, C. (2005). Relations in biomedical ontologies. *Genome biology*, **6**(5), R46.

W3C OWL Working Group (2009). OWL 2 Web Ontology Language Document Overview. Technical report, World Wide Web Consortium.

Woods, W. A. (1991). Understanding subsumption and taxonomy: A framework for progress. In *Principles of Semantic Networks*, pages 45–94. Morgan Kaufmann.

An ontology-based approach for SNOMED CT translation

Mário J. Silva, Tiago Chaves and Bárbara Simões
Instituto Superior Técnico, Universidade de Lisboa and INESC-ID, Portugal

ABSTRACT

SNOMED CT is a comprehensive multilingual class hierarchy of medical terms used in clinical records. Few translations are available, but, as new concepts and revisions are continuously being added, the manual translation and revision of the terms will remain a major endeavour. We propose a new approach for translating SNOMED CT terms (or named entities) using ontology mapping methods and various existing multilingual resources with translated concepts. Our purpose is generating initial candidate translations, already close to those proposed by medical experts, to be later used in a curated translation process. Our method for automatically translating SNOMED CT is being developed for Portuguese, using DBPedia, ICD-9 and Google Translate as sources of candidate translations of the clinical terms, which could later be verified. Initial results, using a manually translated Portuguese catalog of allergies and adverse reactions (CPARA) to SNOMED CT as ground truth, show that it has high potential.

1 INTRODUCTION

SNOMED Clinical Terms[1], or SNOMED CT, is a comprehensive multilingual class hierarchy of terms used in clinical records, with extensive overlapping and synonymous descriptions. The primary purpose of SNOMED CT is to encode the meanings of the terminology used in health information, supporting the effective clinical recording of data with the aim of improving patient care. SNOMED CT provides the core general terminology for electronic health records. With about 300,000 active terms, SNOMED CT spans clinical findings, symptoms, diagnoses, procedures, body structures, organisms and other etiologies, substances, pharmaceuticals, devices and specimen.

The need to interchange medical records across states is demanding the development of faster methods to obtain approved, standards-based, translations of medical records, in particular SNOMED CT. The standardisation of clinical terms and their translations to other languages is very important for the unification of the electronic health records worldwide. However, the manual translation and revision of the terms, synonyms and definitions to a new language is a major endeavour. SNOMED CT is presently available in US and UK English, Spanish, Danish and Swedish. It is also being translated to several other languages, but there is no translation of SNOMED CT to Portuguese or an official initiative to develop and maintain that translation. Hence, a tool to automatically translate SNOMED CT to Portuguese would assist in the production of a release to be validated and improved in a subsequent step at a much lower cost than conducting the process manually.

As new trannslations, concepts and revisions are continuously being added, the manual translation and revision of the terms will remain a major endeavour. This paper describes our work on the development of an automatic translator of SNOMED CT to Portuguese as an assisting tool that could be used for the production of a future standard translation of SNOMED CT. We take the approach of using available classifications and automatic translation services as ontologies that can be aligned and later navigated to provide the translations of such technical terms. In our method, we start by identifying existing alignments between SNOMED CT and other selected ontologies, including the releases of SNOMED CT in different languages. For the Portuguese translation, given its proximity to Spanish, many terms in the Spanish release of SNOMED CT have almost identical spelling. There are other medical terminologies for which multiple translations have been published, such as ICD (International Classification of Diseases) [2]. Another major resource is DBPedia, an ontology derived from Wikipedia, which is very rich in medical terms (Lehmann *et al.*, 2015). After the collection of these multilingual ontologies and published mappings between their terms, we derive additional alignments using the ontology mapping algorithms implemented in AgreementMakerLight, a scalable automated ontology matching system developed primarily for the life sciences domain Faria *et al.* (2013). To obtain correspondences between terms in two distinct languages, we can also explore online translation services, such as the Google Translate Service[3] or Microsoft translator[4] to generate additional mappings.

To show the feasibility of the above outlined approach for automatically generating translations of SNOMED CT terms to Portuguese, we evaluated the translations obtained with the alignments against the translations of a set of SNOMED CT terms that have been mapped by medical experts to terms of the Portuguese catalog of allergies and adverse reactions (SPMS, 2015). The latest release of CPARA includes curated translations of SNOMED CT terms. The evaluation shows promising results. The ontology-mapping translation method achieved an accuracy of 89% and coverage of 37% for the set of 191 terms on the translation of the CPARA vocabulary terms previously hand-mapped to SNOMED CT (using case-insensitive string comparison).

2 RESOURCES AND RELATED WORK

In our work, we used the the 01/2014 International (English) distribution of SNOMED CT and the Spanish version dated from April 2014, both provided by the NLM (National Library of Medicine) institutional site [5]. The distribution also includes a mapping between ICD-9, a WHO classification of diseases, and SNOMED CT. This mapping can be used to link SNOMED CT

[1] http://www.ihtsdo.org/snomed-ct

[2] http://www.who.int/classifications/icd/en/

[3] https://translate.google.com/

[4] http://www.microsoft.com/translator/translator-api.aspx

[5] http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html

codes to the ICD-9 Portuguese terms in a translation provided by the Portuguese Ministry of Health [6].

There are no comprehensive medical terminologies for European Portuguese. In addition to ICD-9 in European Portuguese, ICD-10 has been manually translated to Brazilian Portuguese [7]. There is also an English to Brazilian Portuguese dictionary of medical terms(Stedman, 2003). However, there are a number of terminological differences between these two variants of the language. Other terminologies, such as ICPC [8], have been translated [9], but they have a much narrower scope than ICD.

In computing, the translation of a terminology, such as the set of SNOMED CT terms, is an instance of a common task in Natural Language Processing (NLP) , designated as Named Entity Translation Ling *et al.* (2011). The task is formulated as the problem of, given a set of labels (named entities) in a source language, obtaining the translations of these entities in a target language. Langlais *et al.* (2008) researched the translation of medical terms using a bilingual lexicon. Recently,Abdoune *et al.* (2013) performed an automatic translation of the CORE subset of SNOMED CT to French by mapping this subset to four French-translated terminologies integrated in the UMLS Metathesaurus: SNOMED international, ICD10, MedDRA and MeSH. They were able to map 89% of the preferred terms of the CORE Subset of SNOMED CT with at least one preferred term in one of the four terminologies.

Other approaches for generating translations have been attempted. Algorithms based on linguistic rules are particularly useful for languages which are poor in language resources, like a recently proposed Basque semi-automatic translation of SNOMED CT (Perez-de Viñaspre and Oronoz, 2014). The algorithm takes an incremental approach: first a lexical translation is attempted; then if a translation is not found, generation/transcription-rules for terms, or chunk-level generation to translate a term token by token are used; finally, a rule-based automatic translation system is used to find a translation.

In this work, we explore DBPedia, an ontology derived from Wikipedia, as an alternative source of term translations (Lehmann *et al.*, 2015). We apply ontology matching methods to align DBPedia and SNOMED CT, along with other web-based services, like Google Translate. The DBPedia is a potentially rich resource for medical terms mappings, given that the English and Portuguese Wikipedias are among the largest. To map these ontologies we used AgreementMakerLight, an ontology matching system developed to tackle large ontology matching problems, and focused in particular on the biomedical domain (Faria *et al.*, 2013). This system can handle the mapping of very large ontologies, as it is the case with SNOMED CT and DBPedia. AgreementMakerLight is derived from the AgreementMaker ontology matching systems (Cruz *et al.*, 2009). The alignments produced by AgreementMaker combine multiple matching algorithms, in three layers: the first layer uses string matching methods to identify similar labels, the second matches ontology structures, and the third layer combines the results from the matchers in the first two layers. The initial experiments reported in this paper only used the first layer algorithms to perform the alignments.

Medical terms, like named entities in general, can matched using similarity metrics like the Jaro distance, initially proposed for record linkage systems (Porter and Winkler, 1997). The Jaro distance has been used for the evaluation of automatic translations of named entitites. It accounts the number of transpositions between two input strings and also the number of different characters, resulting in a numeric distance in the $[0, 1]$ range.

3 SNOMED CT TRANSLATION

Given that SNOMED CT is mostly used to provide terminology for electronic health records, the risks of using an automatically generated translation of such large collection of terms without expert validation are unacceptable. In fact, the SNOMED publisher provides detailed guidelines for validating the translations made by medical experts for the official translations available (IHTSDO, 2012). However, we believe that, if the initial quality of the automatically generated translation is high, we could later validate such candidate translations through a crowdsourcing activity, as experimented by Schulz *et al.* (2013).

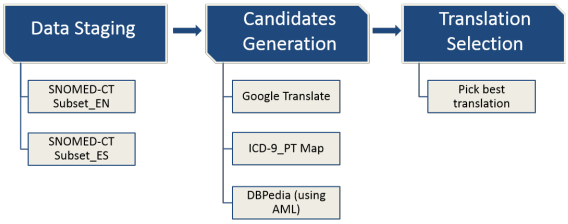


Fig. 1. The translation of SNOMED CT is preceded by a data staging phase. Once the data is prepared, translation is carried out using the implemented methods. We select the best translation candidate using an ensemble model trained that selects the best method for each class of SNOMED CT terms, based on known translations

Our approach for generating the translations of SNOMED CT terms into Portuguese is illustrated in Figure 1. We start by organising two mappings:

1. SNOMED CT to ICD-9: a correspondence between the codes of SNOMED CT and codes and descriptions of ICD-9.
2. SNOMED CT to DBPEDIA: a correspondence between SNOMED CT codes and DBPedia (English and Portuguese) page URIs, and associated page titles.

The first mapping is derived the SNOMED CT to ICD-9 mapping included in the from the UMLS distribution, which includes the correspondence between SNOMED CT and ICD-9 codes. For the second mapping, the matching algorithms implemented in AgreementMakerLight can generate an alignment between SNOMED CT terms and English DBPedia labels. Once this alignment is generated, we can map SNOMED CT codes to DBPedia URIs and then obtain the corresponding label for the Portuguese term by a simple lookup.

To obtain candidate translations for SNOMED CT terms, we implemented four translation methods:

[6] http://www.acss.min-saude.pt/Portals/0/ICD9CMOut2013.xlsx

[7] http://www.datasus.gov.br/cid10/V2008/cid10.htm

[8] http://goo.gl/IX9mqT

[9] http://icpc2.danielpinto.net/

1. Google Translate EN: the candidate translation into Portuguese of each English term in SNOMED CT is provided by the GoogleTranslate API service.
2. Google Translate ES: identical to the above, but the translation service uses the Spanish term as input.
3. ICD-9 Mapping: for a given SNOMED CT term in English, we take the corresponding code and lookup the SNOMED CT to ICD-9 mapping in the UMLS distribution to obtain the ICD-9 code and next the term description in the Portuguese version of ICD-9. This description becomes the candidate translation of the SNOMED CT term to Portuguese.
4. DBPedia Mapping: starting with a SNOMED CT term in English, we lookup the code on the SNOMED CT to DBPedia mapping and, from there, obtain the available candidate translation on the Portuguese DBPedia.

DBPedia is too big to be fully mapped in one batch with limited computing power, given the size of the ontologies involved. This would make the time required by AgreementMakerLight to align SNOMED CT with the full DBPedia prohibitive. However, it is unnecessary, given that most of DBPedia is irrelevant to the clinical domain, to use the full DBPedia. We expect that our users, domain experts in clinical specialisations, will select a batch of SNOMED CT terms of their interest at a time and create/revise the translations of the terms in that smaller set. For instance, to identify a set of allergy-related DBPedia pages to be aligned with a set of SNOMED CT terms, we used the UNIX grep tool to filter out of the DBPedia ontology every page with a label not containing any of the words of the SNOMED CT terms. This resulted in a size reduction from 2 GB to 12MB. To obtain the alignment with DBPedia, we parameterized AgreementMakerLight to consider as aligned all pairs of terms with a Jaro Distance ≥ 0.5.

The last step in our method involves the application of an ensemble learning algorithm (Dietterich, 2000). Each SNOMED CT term has a class label, provided as "qualifier" in the term description. For instance, the SNOMED CT term with code `158965000` has the term "Medical practitioner (occupation)", from which we can separate the description "Medical practitioner" and class *Occupation*. Instead of choosing the best overall translation method, we identify the best translation method for each class, based on the validated translations. As this number will increase over time, we expect that ensemble learning will in the end improve the automatic translation process. However, given the small number of validated and translated terms in Portuguese that we have at this time, we still lack reliable data to evaluate this step.

4 EVALUATION

CPARA, *Catálogo Português de Alergias e Reacções Adversas*, is a list of terms related to Allergies and Adverse Reactions in use in the Portuguese National Health Service (SPMS, 2015). It was developed with the goal of unifying the classification for allergies and adverse reactions in Portugal. Given the high levels of patient mobility, physicians frequently need to know precisely which substances are known to affect an international patient. To address this need, CPARA terms have been mapped to SNOMED CT terms by a group of experts. These experts also created European Portuguese translations of the SNOMED CT Common Terms and Fully Specified Names (FSN) in the CPARA catalog. This mapping is critical to making the medical information exchanged about patients who travel internationally more accurate. In our evaluation, we used the Common Terms translations as gold standard to assess the accuracy of our translation approach. CPARA includes 191 codes and common terms of the US English distribution of SNOMED CT, and the corresponding CPARA codes and terms. In the Spanish SNOMED CT distribution there are 192 terms mapped from these 191 codes (one code is mapped to two terms).

Evaluation of the translated SNOMED CT terms started with candidate translations for the allergy-related SNOMED CT codes in CPARA generated by application of our method. We then evaluated the resulting set of translations against the ground truth composed by the corresponding CPARA terms as defined by the medical committee that defined the mapping. To assess the accuracy of the evaluated translation methods, we scored each term translation by the Jaro distance between the automatically translated term and the CPARA translation. The Jaro distance (JD) between two strings is 1 if the strings have the exact same number of characters and do not have any transposition-[10].

Prior to computing Jaro distances all the translation candidates and CPARA translations were normalised: we removed any qualifiers from the SNOMED CT candidates, deleted quotes from the CPARA translations, and converted all the named entities to lowercase (e.g, "Moderate (severity modifier) (qualifier value)" became "moderate" and "Contact metal agent (substance)" became "contact metal agent"). These preparatory steps are necessary to obtain meaningful similarity metrics, because these qualifiers are common to many terms and can be translated independently. In addition, the Jaro Distance considers the same letter in lowercase and uppercase forms as two distinct characters. The statistics of the translations obtained with each of the four implemented methods described in the previous section are given in Table 1. In these statistics, we considered as valid the translations with $JD = 1$.

Method	Source Language	Coverage	#Method	AVG JD	STDEV JD
GT	EN	100%	191	0.78	0.22
GT	ES	114%	218	0.58	0.15
ICD 9	EN	10%	20	0.61	0.12
DBPedia	EN	37%	70	0.89	0.03

Table 1. Global Results for all translation methods with the respective average Jaro Distance (AVG JD) and Standard Deviation Jaro Distance (STDEV JD). The implemented methods are Google Translate (GT), both from English (EN) and Spanish (ES) to Portuguese, ICD-9 Mapping (ICD 9), and DBPedia Mapping (DBPedia). All translations were attempted with two source languages, English (EN) and Spanish (ES). The number of terms translated by each method is given in the # Method column.

We observe that the SNOMED-DBPedia alignment obtains, for a coverage of 37%, both the highest similarity (0.89) and lowest standard deviation (0.03). This shows that we have been able to accurately translate a set of SNOMED CT terms to Portuguese, using basic alignment techniques, through the SNOMED CT to DBPEDIA alignment. However, the generation of translations

[10] The computation of the Jaro distances was made with the Python Jellyfish library https://pypi.python.org/pypi/jellyfish

based on ontology alignments as proposed in this paper also has limitations. In particular, only a fraction of the translations can be obtained by this method, while Google Translate always proposed a translation. Our success with Portuguese may not be granted when aligning SNOMED CT with DBPedia in other languages with smaller Wikipedias.

Google Translate EN showed better accuracy than Google Translate ES. This result was not initially expected, because Spanish and Portuguese are close languages. This may result from the CPARA terms being originally derived from the English terminology. The number of translations obtained with Google Translate ES is higher than the number of terms in the CPARA dataset (yielding the 114% coverage). This is the result of how we have obtained the Spanish SNOMED CT candidate terms for translation. We started from the same initial SNOMED CT codes that we used for the English translation and obtained the Spanish codes matching the *concept_id* and *type_id* of the initial English terms. This generated a higher number of ES candidate terms to translate (218) then the initial EN terms (191).

To evaluate which translation method works best for each class of SNOMED CT terms, we measure which translation method performs best in each class. This method is necessary to later model an ensemble learning stage that could pick the best method for each class. To obtain the results, we divided CPARA in classes for translation purposes. These classes were extracted from the qualifiers defined for the SNOMED full specified name terms. We were interested in observing translation performance differences across classes. To measure the differences, we calculated the similarity and standard deviation as above of all the translation candidates in each class. The results are summarised in Table 2.

The SNOMED-DBPedia alignment generates better translations for all classes, except *Person* and *Qualifier Value*. The poorer performance could, however, reflect that only a small number of related identified terms in the allergy domain have been identified for both classes.

Google TranslateES has better average similarity for the *Person* class than Google Translate EN. This shows that the SNOMED CT translation from Spanish could benefit from using the Spanish language distribution for some CPARA translations.

The translations obtained with the ICD-9 mapping translator are worse than obtained by Google Translate (for both languages). This results from ICD-9 being less comprehensive than SNOMED CT. ICD codes mostly diseases, symptoms or causes of death. Therefore, many of the CPARA terms in SNOMED CT were absent in the ICD-9 to SNOMED CT mapping. The results also indicate that, as expected and observed with ICD-9, terminologies of narrower scope are not useful for translating clinical terms through ontology alignment. The ICD-9 mapping is much less successful than other resources, such as DBPedia and Google Translate, which can provide much higher coverage of candidate translations, in many cases while retaining equal or better accuracy. The ICD-9 mapping method generates a high amount of 1-to-many matchings. However, ICD-9 could still be useful in cases where it generates only one matching description, which is usually very accurate and reliable, attending that matchings between ICD and SNOMED CT and the resulting translations are validated by medical experts.

5 CONCLUSIONS AND FUTURE WORK

SNOMED CT is increasingly prevalent in the heath care sector, resulting from the increasing need to exchange medical records in mobile societies. There is also a growing general interest in accessing standardised machine-readable medical records for improving managed heath care and biomedical research.

We introduced a new methodology for translating SNOMED CT terms, which relies primarily on aligning large ontologies, complementing language-based methods that have been proposed before. We prototyped an initial implementation of this methodology, which obtained high coverage and good accuracy, despite only using string matchers for SNOMED CT and DBPedia alignment along with the domain-independent Google Translator. A translation was considered valid when the expert mapping of an allergy-related SNOMED CT term to Portuguese is identical to the obtained using the SNOMED CT to DBPedia alignment. The accuracy under these settings was 37%. This shows that both the English and Portuguese versions of DBPedia are rich and accurately interlink with medical terms. However, the results for Portuguese may not be indicative of how this method would perform on other languages. Portuguese is one of the top-10 Wikipedia languages in terms of the total number of entries. The coverage of the obtained translations depend on how rich the Wikipedia for a target language is in covering clinical concepts and the extent to which these concepts are mapped to Wikipedia pages in languages for which a SNOMED CT translation exists. In addition, our validation experiment was confined to testing about 200 SNOMED CT Common Terms in the alergies and adverse reactions domain in European Portuguese. It is still unknown how comprehensive and accurate the English and Portuguese Wikipedias are across the full clinical domain, and how this factor affects the accuracy of the SNOMED CT translations.

Some improvements can still be added to the software implementing the presented translation method. For instance, the SNOMED CT to DBPedia alignment should explore the defined semantic relationships between classes and terms in both SNOMED CT and DBPedia. On the other hand, these relationships could be explored to generate accurate translations for untranslated terms in lexical methods to be provided. For this purpose, language resources, such as WordNet, and parallel corpora of named entities, such as previously validated SNOMED CT translations, could be used to learn how words and multi-word expressions should be properly translated.

The measured accuracy of our translation method could still be significantly increased without sacrificing the quality of translations, by relaxing the similarity threshold. The negative impacts of such relaxation are negligible, given that the generated translations will always need to be validated by experts before used in a clinical context. The expert-validation step presently relies on the review of generated translations presented on spreadsheets. A crowdsourcing platform could speed-up the process of creating and maintaining a validated translation of SNOMED CT. Moreover, active learning could also be incorporated in the crowdsourcing platform, leading to fast improvement of the proposed translations as the validated translation can also be used as input to generate good candidates (Ambati *et al.*, 2010).

A complementary assessment of the alignment approach proposed here could be obtained by applying it to the automatic translation with one of the existing released translations, e.g.

Spanish. However, given that we rely on alignments bewteen lexical resources we are not certain if the Wikipedia correspondences between clinical term pages in Spanish and English have been created based on SNOMED CT.

ACKNOWLEDGEMENTS

We thank Daniel Faria and the other members of the SOMER project for help with running AgreementMakerLight and their feedback. We also thank Dr. Anabela Santos for the help with the CPARA translation of SNOMED CT to validate our tool, and Bruno Martins for the pointers to previous works. This work was partially supported by Fundação para a Ciência e a Tecnologia (FCT), grants PTDC/EIA-EIA/119119/2010 (SOMER), UID/CEC/50021/2013 and EXCL/EEI- ESS/0257/2012 (DataStorm).

REFERENCES

Abdoune, H., Merabti, T., Darmoni, S. J., and Joubert, M. (2013). Assisting the translation of the core subset of snomed ct into french. *Studies in health technology and informatics*, **169**, 819–823. DOI:10.3233/978-1-60750-806-9-819.

Ambati, V., Vogel, S., and Carbonell, J. (2010). Active learning and crowd-sourcing for machine translation. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*.

Cruz, I. F., Antonelli, F. P., and Stroe, C. (2009). Agreementmaker: Efficient matching for large real-world schemas and ontologies. *PVLDB*, **2**(2), 1586–1589.

Dietterich, T. (2000). Ensemble methods in machine learning. In *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pages 1–15. Springer Berlin Heidelberg.

Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I., and Couto, F. (2013). The agreement maker light ontology matching system. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences—Confederated International Conferences*, number 8185 in Lecture Notes in Computer Science, pages 527—541. Springer.

IHTSDO (2012). *Guidelines for Management of Translation of SNOMED CT*. IHTSDO - International Health Terminology Standards Development Organisation.

Langlais, P., Yvon, F., and Zweigenbaum, P. (2008). Analogical translation of medical words in different languages. In B. Nordström and A. Ranta, editors, *Advances in Natural Language Processing*, volume 5221 of *Lecture Notes in Computer Science*, pages 284–295. Springer Berlin Heidelberg.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., and Bizer, C. (2015). DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, **6**(2), 167–195.

Ling, W., Calado, P., Martins, B., Trancoso, I., Black, A., and Coheur, L. (2011). Named entity translation using anchor texts. In *The International Workshop on Spoken Language Translation (IWSLT)*.

Perez-de Viñaspre, O. and Oronoz, M. (2014). Translating snomed ct terminology into a minor language. In *Proceedings of the 5th International Workshop on Health Text Mining and Information Analysis (Louhi)*, pages 38–45, Gothenburg, Sweden. Association for Computational Linguistics.

Porter, E. H. and Winkler, W. E. (1997). Approximate string comparison and its effect on an advanced record linkage system. In *Advanced Record Linkage System. U.S. Bureau of the Census, Research Report*, pages 190–199.

Schulz, S., Bernhardt-Melischnig, J., Kreuzthaler, M., Daumkea, P., and Boeker, M. (2013). Machine vs. human translation of SNOMED CT terms. In *MEDINFO 2013*.

SPMS (2015). CPARA – catálogo português de alergias e outras reações adversas / portuguese catalogue of allergies and other adverse reactions. Technical Report V3.0, 09-03-2015, SPMS – Serviços Partilhados do Ministério da Saúde. `http://tinyurl.com/me5jhq7`,`http://tinyurl.com/lehlhaa`.

Stedman, T. L. (2003). *Stedman's English to Portuguese and Portuguese to English Medical Dictionary*. French & European Pubns. ISBN 13: 9780785975281.

Translation Technique	Source Lang.	Class	AVG JD	STDEV JD
Google Translate	EN	Substance	0.82	0.19
		Observable Entity	0.74	NA
		Product	0.96	0.01
		Disorder	0.78	0.25
		Occupation	1.00	0.00
		Person	0.55	0.18
		Qualifier Value	0.79	0.17
		Finding	0.74	0.30
		Event	1.00	NA
		Situation	0.71	NA
		Organism	0.63	0.37
		Severity Modifier	0.79	0.30
		Contextual Qualifier	0.83	0.15
		No Qualifier	0.63	0.28
	ES	Disorder	0.66	0.11
		Substance	0.59	0.14
		Qualifier Value	0.58	0.11
		Contextual Qualifier	0.56	0.08
		Organism	0.62	0.10
		Person	0.57	0.09
		Occupation	0.67	0.04
		Finding	0.67	0.12
		Situation	0.60	0.00
		Observable Entity	0.64	0.00
		Product	0.57	0.03
		Severity Modifier	0.53	0.13
		Event	0.35	NA
		No Qualifier	0.50	0.28
ICD-9	EN	Disorder	0.60	0.12
		Finding	0.68	0.15
DBPedia Matching	EN	Disorder	0.92	0.04
		Substance	0.90	0.03
		Qualifier Value	0.72	0.05
		Event	1.00	NA
		Finding	0.99	0.00
		Organism	0.82	0.09
		Person	0.48	NA
		No Qualifier	0.83	0.09

Table 2. Scores for the different classes of SNOMED CT terms. AVG and STDEV JD column represent the averade and standard deviation of the Jaro Distance; NA indicates that STDEV cannot be obtained because there is only one translation for the class.

Formalization of indicators of diagnostic performance in a realist ontology

Adrien Barton[1,2,*] Régis Duvauferrier[2,3] and Anita Burgun[4]
[1]The Institute of Scientific and Industrial Research, Osaka University, Japan
[2] INSERM UMR 1099, LSTI, Rennes, France
[3] CHU de Martinique, Université Antilles-Guyane, France
[4] INSERM UMR 1138 team 22, Centre de Recherche des Cordeliers, Paris, France

ABSTRACT

We present a formalization of indicators of diagnostic performance (sensitivity, specificity, positive predictive value and negative predictive value) in the context of a realist ontology. We dissociate the indicators of diagnostic performance from their estimations and argue that the former should be represented in a first place in biomedical ontologies. Our formalization does not require to introduce any possible, non-actual entities - like the result a person would get *if* a medical test would be performed on her - and is therefore acceptable in an ontology built in a realist spirit. We formalize an indicator of diagnostic performance as a data item that is about a disposition borne by a group; the diagnostic value of this indicator is given by the objective probability value assigned to this disposition.

1 INTRODUCTION

1.1 Definition of indicators of diagnostic performance

Biomedical ontologies aim at providing the most exhaustive and rigorous representation of reality as described by biomedical sciences. A large part of medical reasoning concerns diagnosis and is essentially probabilistic. It would be an asset for biomedical ontologies to be able to support such a probabilistic reasoning.

Ledley & Lusted (1959)'s seminal article on Bayesian reasoning in medicine defines different kind of probabilistic entities. Consider for example the simple case of an instance of test of type *A* aiming at detecting if a patient in a group g has an instance of disease of type *M*[1]. The performance of test *A* in diagnosing *M* can be quantified by the positive predictive value of this test, hereafter abbreviated PPV, and generally defined as the proportion of people who have the disease among those who would be tested positive by *A* in g (that is, the proportion of true positives among positives); and by the negative predictive value, hereafter abbreviated NPV, and generally defined as the proportion of people who do not have the disease among those who would be tested negative by *A* in g (that is, the proportion of true negatives among negatives). Those two values provide the probability, once the result of test *A* is observed, that the patient has the disease *M*.

[1] These will be abbreviated in the following as "a test *A*" and "the patient has *M*".

However, such positive and negative predictive values are typically not available in the scientific literature. Instead, they are generally computed from other probabilistic values, namely: the prevalence value of *M* in g, generally defined as the proportion of people who have the disease *M* in g, and hereafter abbreviated Prev(g,*M*); the sensitivity value of the test *A* for *M* in g, generally defined as the proportion of people who would get a positive result by *A* among those who have the disease *M* in g (that is, the proportion of true positives among diseased), hereafter abbreviated Se(g,*A*,*M*); and the specificity value of *A* for *M*, generally defined as the proportion of people who would get a negative result by *A* among those who do not have the disease *M* in g (that is, the proportion of true negatives among non-diseased), hereafter abbreviated Sp(g,*A*,*M*). As a matter of fact, these values are related through the following Bayesian equations:

$$\mathrm{PPV}(g,A,M) = \frac{\mathrm{Prev}(g,M)\,\mathrm{Se}(g,A,M)}{\mathrm{Prev}(g,M)\,\mathrm{Se}(g,A,M) + (1-\mathrm{Prev}(g,M))\,(1-\mathrm{Sp}(g,A,M))}$$

$$\mathrm{NPV}(g,A,M) = \frac{(1-\mathrm{Prev}(g,M))\,\mathrm{Sp}(g,A,M)}{\mathrm{Prev}(g,M)\,(1-\mathrm{Se}(g,A,M)) + (1-\mathrm{Prev}(g,M))\,\mathrm{Sp}(g,A,M)}$$

In the wake of Ledley & Lusted (1959), the sensitivity and specificity values have often been considered as depending only on the pathophysiological characteristics of the disease, and thus as independent of the group of people under consideration. However, sensitivity and specificity values do in fact depend upon the group under consideration: this is the "spectrum effect" (Brenner & Gefeller, 1997; for a detailed explanation, see Barton, Duvauferrier & Burgun, 2015). Spectrum effect can be manifested, for example, as a dependence of sensitivity and specificity on the degree of severity of the disease in the group under consideration (Park, Yokota, Gill, El Rassi, & McFarland, 2005).

In the remainder of the articles, sensitivity, specificity, PPV and NPV will be called "indicators of diagnostic performance" and abbreviated "IDPs".

1.2 The challenge of representing indicators of diagnostic performance in an ontology

To the extent that they aim at representing biomedical knowledge and enabling medical reasoning, biomedical ontologies should provide a formalization of IDPs as well as

the prevalence. This article will propose such a formalization in the context of the OBO Foundry (Smith et al., 2007), one of the most massive sets of interoperable ontologies in the biomedical domain, built on the upper ontology BFO.

The question of how probabilistic notions can be represented in ontologies has been tackled from different perspectives in the past. For example, da Costa et al. (2008) have proposed the new PR-OWL format that extends the classical OWL format; we take here a different approach, which does not aim at changing the OWL format. Soldatova, Rzhetsky, De Grave, & King (2013) have described a model in which probabilities can be assigned to research statements. We have proposed an alternative approach (Barton, Burgun, & Duvauferrier, 2012) in which we show how probabilities can be assigned to dispositions, upon which we are going to build here.

Sensitivity and specificity have been recently introduced in the Ontology of Biological and Clinical Statistics (OBCS; Zheng et al., 2014) as subclasses of *Data item* – a classification that we will endorse here, and extend to PPV and NPV. A data item, as defined by the Information Artifact Ontology (IAO), is intended to be a truthful statement about something. In order to formalize IDPs, one should thus clarify what entities in the real world they are about.

Sensitivity value[2], as we said, is generally defined as the proportion of people who would get a positive result by *A* among those who have the disease *M*. But note here the conditional structure: what is referred to is the proportion of true positives among diseased *if* A was performed on them. In practical situations, however, the sensitivity value will be estimated by performing the test on a sample of the population only – not the entire population g. This will lead to two difficulties. First, it will be necessary to differentiate clearly the IDPs' values from their estimations, and to determine which of those should be represented in a first place in an ontology – part 2 will be devoted to this issue. Second, possible-but-non-actual situations cannot be straightforwardly defined in a realist ontology like BFO; this problem will be explained and solved in part 3, by considering that an IDP is a data item about a disposition borne by an instance of group of individuals, whose probability value will be identified to the diagnostic value of the IDP. This will provide a formal characterization of IDPs.

2 THE INDICATORS AND THEIR ESTIMATIONS

2.1 Two limits for the estimations of indicators of diagnostic performance

Numerical estimations of IDPs face two limits (Barton et al., 2015). First, frequencies will be measured on a sample judged to be representative of the population as a whole, and these values are then extrapolated to the frequencies in the entire population. Second, whether a given person has *M* or not cannot generally be known for sure, through reasonable means: sometimes, the only way to be certain is to perform an autopsy on the deceased patient. Consequently, a "gold standard" must be chosen, namely the best reasonable available diagnostic test[3]. If a patient gets a positive result to this gold standard test, one will conclude that he has the disease; if he gets a negative result, one will conclude that he does not have it.

For example, Park et al. (2005) estimate the sensitivity of the Neer test for diagnosing the impingement syndrome; their estimation is made on a sample of 552 patients considered as representative of the general population, using as gold standard surgical observation. The proportion of patients tested positive by the Neer test among *those who are tested positive by surgical operation* in the *sample* is considered as representative of the sensitivity value - which can be interpreted as the proportion of people who would be tested positive by the Neer test among *those who have an impingement syndrome* in the *whole population*. Similar estimation strategies hold for prevalence, specificity, PPV and NPV.

Note that the estimations of the values of the prevalence, sensitivity, specificity, PPV and NPV depend on both the sample and the gold standard; however, the real values of the prevalence, sensitivity, specificity, PPV and NPV, as defined above, depend neither on the sample, nor on the gold standard.

2.2 What should be represented in an ontology?

This being clarified, one can ask which entities should be preferably represented in an ontology: the IDPs' values, or their estimations?

For sure, we have no direct access to such IDPs' values; but this does not imply that they should not be represented in an ontology. To clarify why, consider an analogy: the measure of the ambient temperature by reading the height of a mercury column in a thermometer. Suppose that at a given time, this height is aligned with the sign "20 °C" written on the thermometer. In such a case, an ontology curator would be in a first place interested in formalizing the fact that the ambient temperature is 20°C, rather than in formalizing the fact that the mercury column in the thermometer is at the same height as the sign "20°C".

In a similar fashion, imagine that 65% of people are tested positive for a gold standard of *M* in a sample s of a population g. The ontology should then formalize in a first place the fact that 65% of the people in g have *M*, rather than the

[2] Note the distinction between a sensitivity and its value: a sensitivity is a data item, but its value is a number.

[3] Even if the gold standard consists in the naked-eye observation of a macroscopic disorder associated exclusively with this disease, this can still theoretically lead to a diagnostic error: any empirical evidence is defeasible.

fact that 65% of the people in s have a positive result to this gold standard. This estimation of this prevalence value may be false (it is indeed very likely to be false, strictly speaking), but future estimations will lead to its being corrected to bring it closer to the real value. As a matter of fact, realist ontologies are built according to a fallibilist methodology (Smith & Ceusters, 2010): they represent the state of the world according to our best knowledge at the present instant, and can be corrected as our knowledge of the world is refined.

That being said, it is possible to represent in an ontology the measurement process of a temperature involving the height of a mercury column in a thermometer. Similarly, one could represent the different estimation processes of the IDPs, and the results to which they led. Such processes are biomedical investigations, and should therefore be formalized in an ontology like OBI (Ontology for Biomedical Investigations, Brinkman et al., 2010), a prominent OBO Foundry candidate dedicated to these issues. This would be relevant in order to formalize in an ontology different estimations given by various samples and gold standards. However, medical practitioners are first and foremost interested in the IDPs' values themselves, rather than in their estimations, and thus we will deal here with the formalization of the former.

This clarification being made, we can now consider the second difficulty mentioned at the end of part 1, namely the formalization of possible-but-non-actual situations in BFO.

3 A FORMALIZATION OF INDICATORS OF DIAGNOSTIC PERFORMANCE IN APPLIED ONTOLOGIES

Sensitivity value has been interpreted as the proportion of people who would get a positive result to *A* among *M*'s bearers in *g*. This definition thus involves the condition of performing the test *A* on the members of g. As we said, such a condition is never realized, because the test is performed (at best) on a sample of the population, not on the whole population g: the performance of test *A* on *g*'s members is a *possible* (leaving aside practical difficulties), *non-actual* condition. Interpreting specificity, PPV, and NPV along the former lines would also imply such possible, non-actual conditions.

However, BFO is built according to the realist methodology, which implies that all the instances it recognizes should be *actual* entities (cf. Smith & Ceusters, 2010). Thus, one cannot represent directly such a possible-but-not-actual condition in an ontology based on BFO. In order to solve this difficulty, we will introduce a strategy named "randomization", enabling to formalize the probabilities of interest as assigned to an actual entity, namely a disposition. This strategy will enable to represent IDPs in a realist fashion, compliant with BFO's spirit.

3.1 From proportions to objective probabilities: the randomization strategy

We will explain first how the proportion of a subgroup in a group can be formalized as a probability value assigned to a disposition; this will help explaining later how the proportion of a subgroup in a group undergoing a possible, non-actual condition can be formalized along similar lines.

Dispositions are entities that can exist without being manifested; an example of disposition is the fragility of a glass, which can exist even when the glass does not break. We will use Röhl & Jansen's (2011) model of disposition in BFO, which associates to every instance of disposition one or several instances of realizations, and one or several instances of triggers (a trigger is the specific process that can lead to a realization occurring). In this model, the fragility of a glass is a disposition of the glass to break (the breaking process is the realization) when it undergoes some kind of stress (the process of undergoing such a stress is the trigger); this disposition inheres in the glass. Starting with the definition of these entities and their relations at the instance level, Röhl & Jansen proceed to formalize them at the universal level. We have shown in a former article (Barton, Burgun & Duvauferrier, 2012) how to adapt this model to probabilistic dispositions. Thus, an instance of balanced coin is the bearer of a disposition instance to fall on heads (the realization process) when it is tossed (the trigger process), to which an objective probability 1/2 can be assigned.

We will now apply this model to the situation at hand. Consider the prevalence Prev(g,*M*), which was defined above as the proportion of bearers of *M* in the actual population g. We can define the disposition $d_{g,M}$, borne by the group g, that a person randomly drawn in g has *M*. More specifically, let's write T_g the process "randomly drawing a person in g", and $R_{g,M}$ the process "drawing by T_g someone who has *M*": the triggers of $d_{g,M}$ are instances of T_g and its realizations are instances of $R_{g,M}$. Following the lines of Barton et al. (2012), one can thus define the probability assigned to the disposition[4] $d_{g,M}$, which is the probability of drawing randomly someone who has *M* in g. This probability is equal to the proportion of individuals who have *M* in g, that is, to Prev(g,*M*): as a matter of fact, if there are *e.g.* 10% diseased people in g, then the probability of drawing randomly a diseased person in g is 10%. Thus, the prevalence value can be identified to the objective probability assigned to the disposition $d_{g,M}$. We name this strategy the "randomization" of the proportion of *M*'s bearers among g.

[4] In Barton et al. (2012), a probability was assigned to a triplet (*d*, *T*, *R*) rather than to a disposition *d*, because we had to take into account disposition that may have several classes of triggers or realizations (that is, multi-trigger and multi-track dispositions, cf. Röhl & Jansen, 2011). However, in the present situation, $d_{g,M}$ is simple-trigger and simple-track: all its triggers are instances of T_g , and all its realizations are instances of $R_{g,M}$. Therefore, the probability value assigned to ($d_{g,M}$, T_g , $R_{g,M}$) can be, for practical matters, assigned directly to $d_{g,M}$.

The randomization strategy may not be necessary to formalize a prevalence, which characterizes a proportion in an actual group, and thus could be formalized as such in an ontology based on BFO. But this strategy can also be applied to proportions of people in groups subject to a *possible, non-actual* condition – and thus, be relevant to formalize sensitivity and other IDPs. As a matter of fact, the sensitivity value Se(g,*A*,*M*) was defined as the proportion of people who would get a positive result to *A* among *M*'s bearers in g. This value can be "randomized" as follows. We can define $d_{Se,g,A,M}$ as the disposition to draw someone randomly who is tested positive by *A*, among the individuals of *g* who have *M*. More specifically, let's define the process $T_{Se,g,A,M}$ as the "performance of test *A* on the individuals in *g*, and random draw of an individual among those who have the disease *M*"[5]; and the process $R_{Se,g,A,M}$ as the "drawing by $T_{Se,g,A,M}$ of someone who got a positive result to *A*". The triggers of $d_{Se,g,A,M}$ are instances of $T_{Se,g,A,M}$, and its realizations are instances of $R_{Se,g,A,M}$. One can then define the sensitivity value Se(g,*A*,*M*) as the objective probability assigned to this disposition $d_{Se,g,A,M,}$: indeed, if there are *e.g.* 15% of the diseased people in *g* who would get a positive result by *A*, then the probability of randomly drawing someone who would get a positive test result by *A* among diseased people in *g* is equal to 15%.

Specificity value can be defined along similar lines, as probabilities assigned to actual dispositions borne by the group g noted $d_{Sp,g,A,M}$ (and similarly for the PPV and NPV). Although $d_{Se,g,A,M}$ and $d_{Sp,g,A,M}$ are both dispositions inhering in g, they have different triggers and different realizations; the process $T_{Sp,g,A,M}$ is the "performance of test *A* on the individuals in g, and random draw of an individual among those who *do not* have the disease *M*" and the process $R_{Sp,g,A,M}$ is the "drawing by $T_{Sp,g,A,M}$ of someone who got a *negative* result to *A*".

3.2 A formal model of indicators of diagnostic performance in ontologies

Let us now consider how to formalize these probability values in ontologies. First, a group g will be considered as any collection of humans (for more on collections, see Jansen & Schultz, 2010). $d_{Se,g,A,M}$ is a disposition individual inhering in the group g; and a probability value can be assigned to this disposition using a datatype property **has_probability_value**. Sensitivity of *A* for *M* in g will be denoted $Se_{g,A,M}$, and following OBCS, it will be defined as a data item. Thanks to our analysis above, we can now answer our original question, and state what this sensitivity is about: $Se_{g,A,M}$ **is_about** $d_{Se,g,A,M}$. We can also introduce a relation **has_diagnostic_value** that relates a sensitivity to its value. In our framework, the (diagnostic) value of a sensitivity $Se_{g,A,M}$ is the probability value assigned to the disposition $d_{Se,g,A,M}$; this can be formalized by writing that if s is a sensitivity, then:

s **has_diagnostic_value** p ⇔ ∃ d ∧ d **is_a** *Disposition* ∧ s **is_about** d ∧ d **has_probability_value** p

As $d_{Se,g,A,M}$ is an individual, it cannot be related directly to the universals *A* and *M*. However, it can be related indirectly to them, by the following formalization. First, $d_{Se,g,A,M}$ can be seen as an instance of a disposition universal symbolized as $D_{Se,A,M}$, which has as trigger the processus universal $T_{Se,A,M}$: "performance of test *A* on the members of a group, and random draw of a person among those who have the disease *M*"; and as realization the process universal $R_{Se,A,M}$ defined as "drawing by $T_{Se,A,M}$ of someone who got a positive result to *A*". We can then introduce two new relations *sensitivity_disposition_of_test* and *sensitivity_disposition_for_disease* (abreviated as *se_of_test* and *se_for_disease*) such that $D_{Se,A,M}$ *se_of_test A* and $D_{Se,A,M}$ *se_for_disease M*. These two relations are introduced for pragmatic reasons of facility of use: on a foundational level, $D_{Se,A,M}$ and *M* (resp. *A*) could be related through a complex array of relations and entities that involve the relation *has_trigger* between $D_{Se,A,M}$ and $T_{Se,A,M}$, as well as a sequence of relations between $T_{Se,A,M}$ and *M* (resp. *A*). Such an analysis would raise theoretical issues though, as instances of $D_{Se,A,M}$ can exist even if no instance of *M* or *A* do exist. We would therefore face here issues similar to the ones addressed by Röhl & Jansen (2011) and Schulz et al. (2014).

Finally, we introduce a class *Sensitivity* that can be characterized as a subclass of *Data item*, which is related to a disposition through the above-mentioned relations:

s **instance_of** *Sensitivity* ⇒ s **instance_of** *Data item* ∧ ∃ d **instance_of** *Disposition* ∧ ∃ a **instance_of** *Test* ∧ ∃ m **instance_of** *Disease* ∧ s **is_about** d ∧ d **se_of_test** a ∧ d **se_for_disease** m

We can also introduce $Se_{A,M}$, the class of sensitivities of test *A* for disease *M* (in whatever group), which can be formalized as a subclass of *Sensitivity* related to *M* and *A* through the following relations:

s **instance_of** $Se_{A,M}$ ⇒ s **instance_of** *Sensitivity* ∧ ∃ d **instance_of** *Disposition* ∧ ∃ a **instance_of** *A* ∧ ∃ m **instance_of** *M* ∧ s **is_about** d ∧ d **se_of_test** a ∧ d **se_for_disease** m

Figure 1 summarizes this formalization of sensitivity (with universals in boxes, instances in diamonds, and the numerical value assigned by datatype properties in a circle). Specificity, PPV and NPV can be formalized along similar lines, as data items about dispositions related to tests and diseases through relations that could be labeled *sp_of_test*, *sp_of_disease*, *ppv_of_test,* etc.

[5] In general, we cannot determine in practice with certainty which individuals of *g* have *M*, and which do not; but the practical impossibility to realize this trigger does not preclude to define this entity.

4 CONCLUSION

We have thus provided a practically tractable formalization of IDPs in a realist ontology, which clearly dissociates IDPs from their estimations (which are relative to a sample and a gold standard). It also solves the difficulty of considering possible, non-actual conditions in a realist ontology based on BFO.

Note that IDPs raise also other theoretical issues. For example, one may want to aggregate two sensitivity values Se(g,*A*,*M*) and Se(g',*A*,*M*) assigned to two different groups g and g' in order to reach a finer assessment of the sensitivity in a larger group; how to do this is a question for the meta-analyst though, not the ontologist, who is first and foremost concerned with representational issues.

This model could then be extended in three directions. A first step would consist in formalizing the estimations of the IDPs, and their relations to a given sample and gold standard. Second, the relations *se_of_test* and *se_for_disease* could be reduced to basic relations and entities already accepted in the OBO Foundry. Third, it could be used by ontology-based diagnostic systems that would compute positive predictive values or negative predictive values from the prevalence, sensitivity and specificity values; more generally, it could be articulated with medical Bayesian networks.

As it takes into account the dependence of IDPs upon the group of people considered, it has the potential to contribute to the development of precision medicine (Mirnezami, Nicholson & Darzi, 2012), an emerging approach that takes into consideration patients characteristics and dispositions, including individual variability in genes, to offer more personalized preventive, diagnostic and therapeutic strategies.

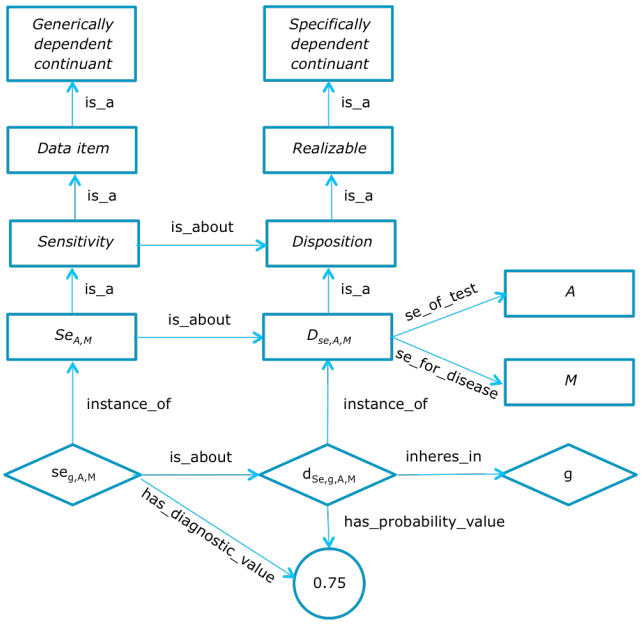


Figure 1 Sensitivity of a test *A* for a disease *M* in a group g with probability value 0.75

ACKNOWLEDGEMENTS

We would like to thank the audience at several seminars, as well as four anonymous reviewers, for their helpful comments. Adrien Barton thanks the Japanese Society for Promotion of Science for financial support.

REFERENCES

Barton, A., Burgun, A., and Duvauferrier, R. (2012) Probability assignments to dispositions in ontologies. *Proc. 7th Int. Conf. Form. Ontol. Inf. Syst. FOIS2012* (M. Donnelly & G. Guizzardi, eds.), 3–14, Amsterdam: IOS Press.

Barton, A., Duvauferrier, R. and Burgun, A. (2015) *Une analyse philosophique des indicateurs de performance des tests diagnostiques médicaux*. Submitted manuscript.

Brenner, H. and Gefeller, O. (1997) Variation of sensitivity, specificity, likelihood ratios and predictive values with disease prevalence. *Stat. Med.*, **16** (9), 981–991.

Brinkman, R. R., Courtot, M., Derom, D., Fostel, J. M., He, Y., Lord, P., Malone, J., et al. (2010) Modeling biomedical experimental processes with OBI. *J. Biomed. Semant.*, **1 Suppl 1**, S7.

Costa, P. C. G. da, Laskey, K. B. and Laskey, K. J. (2008) PR-OWL: A Bayesian ontology language for the semantic web. In: *Uncertainty Reasoning for the Semantic Web I*, 88–107, Springer.

Jansen, L., and Schulz, S. (2011) Grains, components and mixtures in biomedical ontologies. *J. Biomed. Semant.*, **2 Suppl 4**, S2.

Ledley, R. S. and Lusted, L. B. (1959) Reasoning foundations of medical diagnosis. *Science*, **130**(3366), 9–21.

Mirnezani, M.R.C.S., Nicholson, J. and Darzi, A. (2012) Preparing for precision medicine. *N. Engl. J. Med.*, **366**(6), 489-491.

Park, H. B., Yokota, A., Gill, H. S., El Rassi, G., McFarland, E. G. (2005) Diagnostic accuracy of clinical tests for the different degrees of subacromial impingement syndrome. *J. Bone Joint Surg. Am.*, **87**(7), 1446–1455.

Röhl, J., Jansen, L. (2011) Representing dispositions. *J. Biomed. Semant.*, **2** (Suppl 4), S4.

Schulz, S., Martínez-Costa, C., Karlsson, D., Cornet, R., Brochhausen, M., and Rector, A. (2014) An Ontological Analysis of Reference in Health Record Statements. *Proc. 8th Int. Conf. Form. Ontol. Inf. Syst. FOIS2014* (P. Garbacz & O. Kutz, eds.), 289-302, Amsterdam: IOS Press.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., et al. (2007) The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat. Biotechnol.*, **25**(11), 1251–1255.

Smith, B. and Ceusters, W. (2010) Ontological realism: A methodology for coordinated evolution of scientific ontologies. *Appl. Ontol.*, **5**(3), 139–188.

Soldatova, L. N., Rzhetsky, A., Grave, K. De and King, R. D. (2013) Representation of probabilistic scientific knowledge. *J. Biomed. Semant.*, **4**(Suppl 1), S7.

Zheng, J., Harris, M. R., Masci, A. M., Lin, Y., Hero, A., Smith, B., and He, Y. (2014). OBCS: The Ontology of Biological and Clinical Statistics in ICBO2014. *Houston, TX, USA*

Formal representation of disorder associations in SNOMED CT

Edward Cheetham[1], Yongsheng Gao[2], Bruce Goldberg[3], Robert Hausam[4], and Stefan Schulz[5,*]

[1] Health and Social Care Information Centre, UK
[2]International Health Terminology Standards Development Organisation, Copenhagen, Denmark
[3]Kaiser Permanente, USA
[4]Hausam Consulting LLC, Midvale, UT, USA
[5] Institute of Medical Informatics, Statistics and Documentation, Medical University of Graz, Austria

ABSTRACT

Medical terminologies like SNOMED CT often provide codes for frequently co-occurring associations of findings and disorders, such as syndromes or diseases with sequelae. The current release of SNOMED CT still lacks a principled solution for representing these concepts, which was the reason for the IHTSDO project group "Event, Condition, Episode" to elaborate a well-founded approach based on criteria of formal ontology. The group analysed complex SNOMED CT terms and proposes a simple solution, which draws on the interpretation of findings, disorders, and diseases as clinical life phases. Co-occurrence, temporal relatedness and causal relatedness were represented by distinct modelling patterns in OWL-DL.

1 INTRODUCTION

A main purpose of clinical terminologies is to support semantic annotation of the content of medical records. Consequently, in many terminology systems such as ICD-9 and ICD-10, in the draft of the upcoming ICD-11 (WHO, 2015), as well as in SNOMED CT (IHTSDO, 2015), numerous codes denote clinical phenomena that frequently co-occur or are temporally related, so that complex disorders like *Pericarditis with pericardial effusion*, or *Vitamin B12 deficiency anaemia due to malabsorption* can be encoded in one step. Extreme cases are codes for highly specific clinical scenarios like *Extradural haemorrhage following injury without open intracranial wound and with prolonged loss of consciousness (more than 24 hours) without return to pre-existing conscious level*.

Table 1. English UMLS terms containing coordinating and temporal connectors, related to all English UMLS terms.

Substring	Count	Rate
' after '	3,899	0.1%
' and '	337,706	4.7%
' caused by '	3,605	0.0%
' due to '	29,223	0.4%
' with '	231,128	3.2%
' without '	21,131	0.3%
ALL	626,692	8.7%

* To whom correspondence should be addressed: stefan.schulz@medunigraz.at

A review of all English terms in the UMLS Metathesaurus (NLM, 2015), which constitutes the biggest collection of biomedical terminology, reveals the importance of coordinating expressions or particles as parts of domain-specific terms (Table 1). Most of these terms denote findings, events, disorders, and procedures. In SNOMED CT, the picture is similar, as shown in Table 2. Many of these terms had been incorporated into SNOMED CT because of efforts to align ICD 9 and 10 with SNOMED CT.

Table 2. Distribution of coordinating and temporal connectors in English SNOMED CT fully specified names (percentages, rightmost column absolute count). Total number of concepts approx. 300,000.

	' after '	' and '	' caused by '	' due to'	'with'	'without'	Σ	Σ_{abs}
Body Struct.	.0	2.5	.0	.0	.6	.0	3.1	951
Clin. Finding	.1	3.0	.1	2.7	4.7	1.3	11.9	11,974
Event	.6	8.1	12.0	1.9	9.6	3.1	44.3	1,627
Obs. Entity	.3	2.3	.0	.0	.5	.0	3.1	257
Product	.0	1.5	.0	.0	.0	.0	1.5	259
Phys. Object	.0	2.3	.0	.0	5.4	1.2	9.0	408
Procedure	.1	5.8	.0	.0	5.8	.3	12.1	6,497
Qual. Value	.2	1.1	.0	.0	.4	.0	1.8	162
Situation	.3	1.8	.0	.2	3.9	.4	6.6	243
Substance	.0	1.1	.0	.0	.4	.0	1.5	353
Others	.0	1.2	.0	.0	.1	.0	1.3	584
ALL	.0	2.9	.2	1.0	3.0	.5	7.7	23,039

While SNOMED CT is increasingly incorporating principles of applied ontology and provides a description logics (DL) (Baader *et al.*, 2007) based version implementing OWL EL (Motik *et al.*, 2012), the current representation of co-ordinating expressions in SNOMED CT does not follow clearly defined patterns. For instance, the definition of the SNOMED CT concept *Diabetic retinopathy* (disorder) uses the relation **associated with** for linking with *Diabetes mellitus*, whereas *Paraneoplastic neuropathy* (disorder) is con-

nected to *Neoplastic disease* using **due to**. Another example is given by the concepts *Dermatomycosis associated with AIDS* (disorder) and *AIDS with dermatomycosis* (disorder), which appear to be duplicates. Whereas the former one uses the relation **associated with** for establishing a connection with the concept *AIDS*, the latter one is represented as a subclass of *AIDS*. This motivated the project group *Event, Condition, Episode Model* (ECE) of IHTSDO[1], the organization that maintains SNOMED CT, to conduct a thorough investigation of this phenomenon and to suggest a solution that is in line with current principles of ontology development in SNOMED CT.

2 METHODS

The ECE group decided to limit the scope of the investigation to the SNOMED CT hierarchy *Clinical Finding* / *Disorder*, following IHTSDO's current strategic directions in the content development process (IHTSDO, 2010). SNOMED CT statements that implicitly include negation were also not considered because they are not expressible in OWL-EL. All group members selected SNOMED CT term samples that represented coordination phenomena, in order to propose recurring modelling patterns. Having done this, the group discussed the underlying meaning, in particular the ontological commitment of the sample *Finding / Disorder* concepts and the underlying semantics with regard to time and causality. As an ontological reference, BioTop-Lite2 (BTL2) (Schulz & Boeker, 2013), an upper-level ontology based on OWL DL and tailored for the biomedical domain, was used. BTL2 provides a small set of upper-level classes, mappable to BFO (2015). All BTL2 classes exhibit a set of constraining axioms using a set of canonical relations, partly derived from the OBO Relation Ontology (Smith *et al.*, 2005). BTL2 heavily constrains the freedom of the ontology engineer, which warrants a higher predictability of the ontologies produced.

Table 3. Four patterns found for "*X* with *Y*"

Pattern	Definition	Example
1	Both *X* and *Y* are co-occurrent, but with no causality or manifestational relationship between *X* and *Y*	*Hay fever with asthma*
2	*X* is due to *Y*, but *X* and *Y* are not necessarily co-occurrent	*Disorder of optic chiasm due to non-pituitary neoplasm*
3	X temporally follows Y. This does not specify that *X* is due to *Y*, although causality is frequently implied	*Postvaricella encephalitis*
4	X is due to *Y*, and both *X* and *Y* are co-occurrent	*Hernia, with intestinal obstruction*

[1] http://www.ihtsdo.org/participate/project-groups

3 RESULTS

3.1 Typology and ontological analysis

Our analysis yielded four patterns of coordinative expressions in the SNOMED CT Finding / Disorder hierarchy, as shown in Table 3.

Further analysis focused on the following questions:

- Which are exactly the entities that are denoted by the concepts under scrutiny?
- Which temporal relationships have to be distinguished?
- What does causality mean and how is it linked to temporality?

According to Schulz *et al.* (2012b), we interpret all finding / disorder codes as *Clinical Situations* or *Clinical Life Phases* (we will use the latter term and illustrate it by the suffix "CLP") i.e. a patient's life phase during which a clinically relevant condition is present. For instance, the SNOMED CT concept $Encephalitis_{CLP}$ denotes the class of processual entities of the type *Life phase*, in which some encephalitis process is present in any temporal instant covered by this life phase. Accordingly, $Hernia_{CLP}$ denotes the class of processual entity of the type *Life phase*, in which the material disorder *Hernia* is fully present. The advantage of this interpretation is that we do not have to deal with hierarchies of entity types of different ontological categories under the same umbrella. To this end, BTL2 provides the defined class *Condition* – the disjunction of *Material object*, *Disposition* and *Process* (Schulz *et al.*, 2011a) – and the class *Situation* as a life phase during which some condition holds: an X_{CLP} is a Life phase during which some condition *X* is fully present. If John has constant headache today from 6am to 11pm, this period of his life is of the type $Headache_{CLP}$. If he is seen by a doctor between 3pm and 3.10pm, this ten-minute lifespan is a new instance of the same type. If he also suffers from diabetes mellitus, then these life phases also instantiate '*Diabetes mellitus*'$_{CLP}$. We formalize this in OWL DL in the following way, using OWL Manchester Syntax:

X_{CLP} equivalentTo '*Clinical life phase*' and
'**has condition**' some *X*

The relation '**has condition**' in BTL2 holds between a life phase and an entity that is constantly present during this life phase. Independently, we have to look at temporality, where we need to clarify what "following" and "co-occurring" exactly mean. In BFO2 we find the relation '**is preceded by**', which is defined as relating two processes, one of which ends before the second one begins.

A commonly accepted framework for describing temporal relations is Allen's (1983) interval calculus (Fig. 1). Compared to this, a relational statement based on BTL2 "x **is preceded by** y" corresponds to either the Allen-based statement "y **takes place before** x" or "y **meets** x".

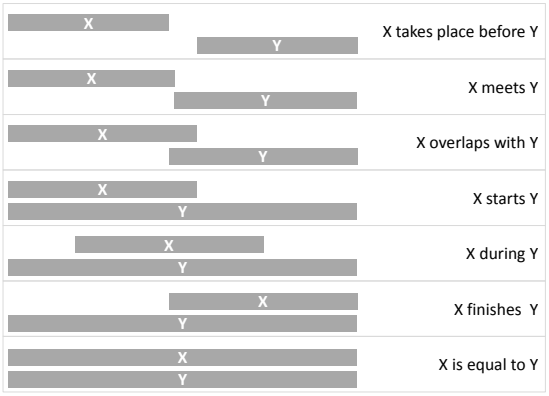


Fig. 1. Base relations of Allen's interval calculus. The converse relations are not depicted

Looking at the examples where we had asserted co-occurrence, we agreed to interpret "x **co-occurs with** y" as the disjunction of "x **starts** y", "x **during** y", "x **finished** y", and "x **is equal to** y". Let us take the example *Hay fever with asthma* (Fig. 2). We have three *Clinical life phase* entities: *'Hay fever with asthma'*$_{CLP}$, *'Hay fever'*$_{CLP}$, and *Asthma*$_{CLP}$. The possible temporal patterns result from any combination of Fig.2 left hand side with Fig.2 right hand side. All temporal instants of Hay fever with *Asthma*$_{CLP}$ temporally coincide with some instant of *'Hay fever'*$_{CLP}$ and some instant of *Asthma*$_{CLP}$.

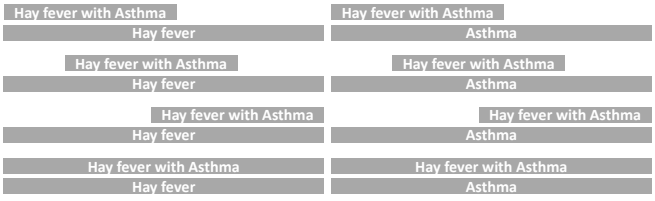


Fig. 2. Co-occurrence of combined situations

Finally, we will have a look at causality. Notwithstanding past and on-going philosophical debates about its nature, we consider the notion of causality as a primitive predicate, which is essential for medical reasoning and decision-making. Whether y follows x accidentally or because it is caused by x is seen as fundamentally different. There are important temporal implications of causality. It is a truism that an effect cannot precede its cause, or conversely, an effect has to follow its cause. Referring to the Allen calculus, "x **causes** y" would then be only compatible with "x **takes place before** y", "x **meets** y", "x **overlaps** y" as well as with (switching the arguments x and y) "y **during** x" and "y **finishes** x". All these relations have in common that the starting point of x precedes the starting point of y.

3.2 Proposal of modelling patterns

In the following, we will propose modelling solutions for *'X with Y'* concepts in SNOMED CT for the frequently encountered clinical patterns in Table 3. The simplest modelling approach for representing X with Y concepts in SNOMED CT is:

1. Both *X* and *Y* are co-occurrent, but with no causality between *X* and *Y*.

X_{CLP} with Y_{CLP}, both simply asserted as co-occurring, and no known causal/manifestational relationship implied:

XwithY$_{CLP}$ equivalentTo X_{CLP} and Y_{CLP}

XwithY$_{CLP}$ denotes the class of life phases that are characterised by the full presence of both the conditions *X* and *Y*:

XwithY$_{CLP}$ equivalentTo '*Clinical life phase*' and '**has condition**' some *X* and '**has condition**' some *Y*

It can be shown that both definitions are equivalent, by logical transformation or by a reasoner like HermIT (2015). An important result (as represented by the second definition) is that for each time interval [t_1; t_2] any single human is considered to have one single life phase, which is characterised by the conditions that are wholly present in this interval. As discussed in Schulz *et al.* (2011b), the subsumption of complex disorder classes by their constituent disorder classes is a characteristic phenomenon in many disease / disorder terminologies, and the life phase interpretation puts it on a solid ground.

It can easily be shown that the same applies for complex clinical life phase types with more than two conjoints, e.g. in the case of the Tetralogy of Fallot (Schulz *et al.*, 2011b), a combined heart defect as an emblematic example.

2. *X* is due to *Y* but *X* and *Y* are not necessarily co-occurrent

Here, the correct way would be to assert causality between the conditions *X* and *Y*.

XcausedByY equivalentTo *X* and '**is caused by**' some *Y*

However, according to our interpretation, SNOMED CT disorder concepts are clinical life phases and the underlying

conditions are not or are only indirectly available[2]. We therefore axiomatically extend the notion of causation and allow that a clinical life phase is causally related to another clinical life phase. To express this, we use the SNOMED CT relation '**due to**'.

> *XcausedBy*Y_{CLP} equivalentTo X_{CLP} and
> '**due to**' some Y_{CLP}

This is a simplification of the correct representation, which should be (using the BTL2 relation '**is caused by**'):

> **XcausedBy*Y_{CLP} equivalentTo '*Clinical life phase*' and
> '**has condition**' some (*X* and '**is caused by**' some *Y*)

Due to the problem of referring to clinical conditions in SNOMED CT, in the modeling pattern we propose, '**due to**' connects two CLPs that are related by the fact that the first one has a condition that is caused by a condition that defines the second one. E.g., all instances of '*Disorder of optic chiasm due to non-pituitary neoplasm*'$_{CLP}$ imply an instance of '*Non-pituitary neoplasm*'$_{CLP}$ (which implies an instance of '*Non-pituitary neoplasm*'). We consider this approximation as sufficient for the reasoning services required.

3. *X* temporally follows *Y*. This does not specify that *X* is due to *Y*, although causality is frequently implied

> *XfollowsY*$_{CLP}$ equivalentTo X_{CLP} and
> **follows** some Y_{CLP}

As mentioned before, the BTL2 relation **is preceded by** excludes the Allen relation **overlaps**. We argue that the relation **follows** should include **overlaps**, as it is common in medicine. E.g., *Postvaricella encephalitis* might include cases in which the varicella infection has not ended at the inception of the complication, viz. *Encephalitis*.

4. *X* is due to *Y* and both *X* and *Y* are co-occurrent
Here we propose a combination of patterns 1 and 2:

> *XdueToCooccurringY*$_{CLP}$ equivalentTo
> X_{CLP} and Y_{CLP} and '**due to**' some Y_{CLP}

[2] In the case of fully defined disorder concepts, the condition would correspond to the combination of location with morphology, inside the role groups

It looks uncommon that the same class Y_{CLP} appears both as a superclass and a class related via the relation **due to**. Taking the example '*Hernia with intestinal obstruction*'$_{CLP}$: All life phases of this type are both *Hernia* life phases and *Intestinal obstruction* life phases, and they are related, additionally, to a second *Hernia* life phase (which is a different one but is assumed to refer to the same hernia object). This second life phase is, actually, one that precedes the inception of the complication, in this case the intestinal obstruction.

3.3 Special cases

There are other cases that we have excluded from our typology, but which, nevertheless, deserve consideration:

- Terms of the type '*Abscess of urethral gland due to Neisseria gonorrhoeae*'. Here, the right hand side of the particle "due to" denotes a material agent, not a process. In BTL2 this would be expressed by the relation '**has agent**' and could be modeled in the following way (note that in this case, the relation '**has condition**' is a paraphrase of the role group relation in SNOMED CT):

  > *XwithAgentA*$_{CLP}$ equivalentTo X_{CLP} and
  > '**has condition**' some ('**has agent**' some *A*)

- Associativity. It can be shown that the following rule always holds and can be easily reduced to pattern one:

  > *XwithY*$_{CLP}$ and Z_{CLP} equivalentTo
  > X_{CLP} and *YwithZ*$_{CLP}$ equivalentTo
  > X_{CLP} and Y_{CLP} and Z_{CLP}

  E.g. '*Diabetes mellitus with hyperosmolar coma*'$_{CLP}$ superficially appears to be an *X* with *Y* pattern, but is more appropriately an example of *X* with *Y* with Z (*Diabetes mellitus*, *Hyperosmolar state*, *Coma*). There is no nesting.
- On this basis, more complex chained sequences according to pattern four are possible using the '**due to**' relationship. This might be represented as:

  > '*Diabetes mellitus with hyperosmolar coma*'$_{CLP}$
  > equivalentTo
  > '*Diabetes mellitus*'$_{CLP}$ and
  > ('*Hyperosmolar state*'$_{CLP}$ and
  > '**due to**' some '*Diabetes*'$_{CLP}$) and
  > (*Coma*$_{CLP}$ and
  > '**due to**' some '*Hyperosmolar state*'$_{CLP}$)

4 DISCUSSION

Table 4 gives an overview of the relations used in the proposed models of our approach and their mapping to the Allen relations. As the BTL2 relation '**is preceded by**' does not allow overlap, it seems too strict. We prefer the relation **follows**, which makes the minimal assumption that the beginning of *Y* is later than X. This is also the assumption of the causality relation, which appears as a subrelation of **follows**. The proposal to include an overlap pattern for temporally following again blurs the distinction between pattern two and pattern three. This might be acceptable if we do not want to distinguish sequelae from other types of complications. When investigating definitions of sequelae under the concepts *Sequela* (finding) or *Sequela of disorders* (disorder), we found chronic or residual conditions that are complication of acute conditions that occur after the acute disease or injury phase. Sequelae can also be the result of the treatment of the primary condition. There is no time limit on when a late effect can occur; the residual condition may come directly after the disease or condition, or years later. This is a little vague in terms of whether the inciting condition is still present when the complication commences. In case there is a requirement to represent sequelae (late effects) as distinct from e.g., immediate complications, it might be worthwhile to define sequelae as not overlapping with their cause, and for this case to indeed use the BTL2 relation '**is preceded by**'.

Table 4. Allen relations compatible with the relations used in our models.

	Proposed Relations			
Allen Relations	Y co-occurs with X	Y is preceded by X	Y follows X	Y is due-to X
X takes place before Y		√	√	√
X meets Y		√	√	√
X overlaps with Y			√	√
Y starts X	√			
Y during X	√		√	√
Y finishes X	√		√	√
X is equal to Y	√			

5 CONCLUSION AND FURTHER WORK

The proposed patterns have been prototypically implemented in SNOMED CT and have achieved better semantic clarity and consistency in terminology creation and maintenance. The formal analysis of temporal and causative relationships has been proved to be useful for determining the patterns. Limitations identified and resulting tasks will be addressed by the ECE working group in the future:

- To evaluate if the proposed patterns are generic and can be applied throughout SNOMED CT, especially to concepts in the *Event* and *Procedure* hierarchies.
- To prove theoretically and empirically that the proposed patterns do not produce unexpected classification results, especially as a consequence of the simplification by asserting '**due to**' between situations and not between the underlying conditions.
- To check the impact of the new models on classification time.
- To extend the approach to negated conditions ("without") by scenarios that extend DL expressiveness or that represent negations as primitives. The impact on reasoning behaviour will also be investigated.
- To propose adjustments to the SNOMED CT naming conventions in the light of the new model.

REFERENCES

Allen, J. F (1983): Maintaining knowledge about temporal intervals. *Communications of the ACM* **26**(11), 832–843.

Baader, F. et al. (2007). The Description Logic Handbook. Cambridge: Cambridge University Press.

BFO (2015) Basic Formal Ontology. http://ifomis.uni-saarland.de/bfo/

HermiT (2015). HermiT OWL reasoner. http://www.hermit-reasoner.com/

IHTSDO (2010). Strategic Directions to 2015. http://www.ihtsdo.org/resource/resource/1

IHTSDO (2015) International Health Terminology Standards Development Organisation. SNOMED CT.

Motik, B. et al. (2012) OWL 2 Web Ontology Language Profiles (Second Edition). W3C Recommendation http://www.w3.org/TR/owl2-profiles/

NLM (2015). U.S. National Library of Medicine. Unified Medical Language System (UMLS), http://www.nlm.nih.gov/research/umls

Schulz, S. et al. (2011a). Scalable representations of diseases in biomedical ontologies. Journal of Biomedical Semantics. May **17**; 2 Suppl 2: S6.

Schulz, S. & Boeker, M. (2013). An Upper Level Ontology for the Life Sciences. Evolution, Design and Application. In: Furbach, U. & Staab, S. (eds.). Informatik 2013. IOS Press.

Schulz, S. et al. (2011b). Consolidating SNOMED CT's ontological commitment. Applied ontology **6**: 1-11.

Schulz, S. et al. (2012b). Competing interpretations of disorder codes in SNOMED CT and ICD. AMIA Annu Symp Proc. 819-827.

Smith, B. et al. (2005). Relations in biomedical ontologies. Genome Biology **6**(5): R46.

WHO (2015). International Classification of Diseases. http://www.who.int/classifications/icd/en/

Can SNOMED CT be Squeezed Without Losing its Shape?

Pablo López-García*, Stefan Schulz

Institute for Medical Informatics, Statistics and Documentation – Medical University of Graz
Auenbruggerplatz 2, 8036 Graz, Austria

ABSTRACT

In biomedical applications where the size and complexity of SNOMED CT are challenging, using a more compact subset that can act as a reasonable substitute is often preferred (e.g., in problem lists, using the CORE problem list subset of SNOMED CT, covering 95% of usage in less than 1% its size). Ontology modularization is the area of research that studies how to extract such subsets, also called modules or segments. In a special class of use cases including ontology-based quality assurance, scaling experiments for real-time performance, and developing scalable testbeds for software tools, it is essential that modules are representative of SNOMED CT's sub-hierarchies in terms of concept distribution, therefore preserving the original shape of SNOMED CT. How to extract such balanced modules remains unclear, as most previous work on ontology modularization has focused on the opposite problem: on extracting a representative module for a specific domain. In this study, we investigate to what extent extracting balanced modules that preserve the original shape of SNOMED CT is possible by presenting and evaluating an iterative algorithm.

1 INTRODUCTION

The size and complexity of SNOMED CT[1] constitute a problem in many biomedical applications (Pathak *et al.* (2009)). Studies have shown that it is often enough to use a subset of interest instead of the whole SNOMED CT. This is the case of problem lists, where the 16 874 terms of CORE[2] have been shown to cover over 95% of usage (Fung *et al.* (2010)), when tagging medical images (Wennerberg *et al.* (2011)), or when annotating texts from cardiology (López-García *et al.* (2012)).

How to extract such subsets is studied by the area of research of *ontology modularization* (Stuckenschmidt *et al.* (2009)). Ontology modularization techniques are generally focused on obtaining a minimal subset (also called module or segment) that maximally covers a specific domain or that is representative for a particular application. This is the case of the problem list or annotation cases mentioned above, or the study by Seidenberg and Rector (2006), where they described how they extracted a representative segment of the GALEN ontology (Rogers and Rector (1996)) for cardiology using the seed concept *'Heart'* as a signature.

A *signature* is an initial set of concepts (called *seeds*) that bootstraps the modularization process, on which many ontology modularization techniques rely, including graph-traversal (Doran *et al.* (2007); d'Aquin *et al.* (2007); Noy and Musen (2004); Seidenberg and Rector (2006)) and logic-based techniques (Cuenca Grau *et al.* (2008); Grau *et al.* (2009)).

Often, these modules are not *balanced* when it comes to representing the original distribution or shape of sub-hierarchies shown by the original ontology or terminology. For example, in the CORE subset of SNOMED CT, most concepts belong to the *Clinical Finding*, *Procedure*, *Situation with Explicit Context*, and *Event* sub-hierarchies[2]. The opposite case is also possible: in a previous study, we found out that especially when using graph-traversal techniques resulting modules can excessively and uncontrollably grow and spread across sub-hierarchies (López-García *et al.* (2012)).

These results are not surprising, because most prior work on ontology modularization has not focused on preserving the representativity of the sub-hierarchies of the original ontology, so the shape of the original ontology is inevitably lost in the modules.

There is a special class of use cases, however, where it is essential that modules are representative of the sub-hierarchies of the original ontology and therefore show a similar shape, such as:

- In ontology-based quality assurance, where small but representative samples of a huge ontology are to be inspected (Agrawal *et al.* (2012));
- for obtaining a demonstration version that is understandable for users or facilitates visualization;
- for alignment with a highly constrained upper level ontology, such as the Basic Formal Ontology (BFO) (Smith *et al.* (2005)), especially the upcoming BFO 2.0 OWL version, which includes relations, DOLCE (Gangemi *et al.* (2002)) or BioTopLite (Schulz and Boeker (2013)), where reasoning has to be tested on small subsets and in iterative debugging steps;
- for performing scaling experiments for real-time performance of a large OWL DL ontology;
- for the description logics community, who welcomes scalable testbeds for developing tools like editors and reasoners.

To the knowledge of the authors, little research on ontology modularization has focused on extracting balanced modules for such applications, where keeping the original shape of a large ontology such as SNOMED CT regarding sub-hierarchies is a requirement.

In this paper, we study the concept distribution of SNOMED CT's sub-hierarchies and we propose an evaluate an iterative algorithm for extracting balanced modules. Our main goal is to investigate to what extent it is possible to obtain modules that preserve the original shape of SNOMED CT in order to be used in our identified class of use cases.

*Correspondence should be addressed to: pablo.lopez@medunigraz.at

[1] International Health Terminology Standards Development Organization - `http://www.ihtsdo.org/snomed-ct/` (accessed 27 Feb 2015)

[2] The CORE Problem List Subset from SNOMED CT - `http://www.nlm.nih.gov/research/umls/Snomed/core_subset.html` (accessed 27 Feb 2015).

2 SUB-HIERARCHIES OVERVIEW

Table 1 shows the main 18 sub-hierarchies of SNOMED CT and their concept distribution. As can be seen, there are four sub-hierarchies that each contain over 10% of SNOMED CT concepts (*Clinical Finding*, *Procedure*, *Organism*, and *Body Structure*), adding up to over 70% of the concepts. We used the July 2014 International Release of SNOMED CT, and we omitted the metadata concepts sub-hierarchy (SNOMED CT model).

Subhierarchy (Abbreviation)	Concepts	Distribution
Clinical Finding (CF)	100 893	33.57%
Procedure (PR)	53 914	17.94%
Organism (OR)	33 273	11.07%
Body Structure (BS)	30 685	10.21%
Substance (SU)	24 021	7.99%
Pharmaceutical / Biologic Product	16 881	5.62%
Qualifier Value (QV)	9 055	3.01%
Observable Entity (OE)	8 307	2.76%
Social Context (SO)	4 703	1.56%
Physical Object (PO)	4 522	1.50%
Situation with Explicit Context (SI)	3 695	1.23%
Event (EV)	3 673	1.22%
Environment or Geogr. Location (EG)	1 814	0.60%
Specimen (SN)	1 447	0.48%
Staging and Scales (ST)	1 309	0.44%
Special concept (SP)	649	0.44%
Record Artifact (RA)	227	0.22%
Physical Force (PF)	171	0.08%

Table 1. Main sub-hierarchies of SNOMED CT. The metadata concepts sub-hierarchy (SNOMED CT model) was not considered.

As a useful way of visualizing concept distribution and for comparative purposes (see Section 4), the same information is displayed in form of a treemap in Figure 1. The treemap represents SNOMED CT's hierarchical information as a set of rectangles, where the area of each rectangle is proportional to the number of concepts in the sub-hierarchy.

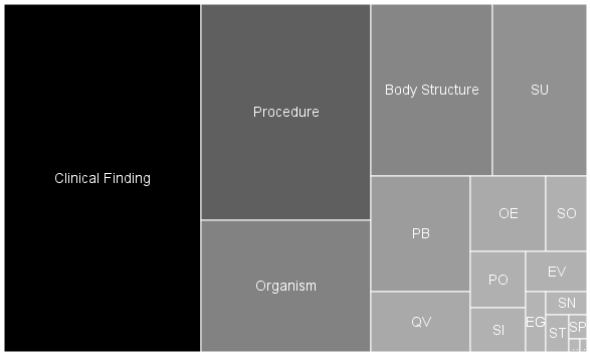


Fig. 1: SNOMED CT's shape represented with a treemap. Sub-hierarchies containing less than 10% of SNOMED CT concepts are shown in acronyms (see Table 1).

3 EXTRACTION OF BALANCED MODULES

As remarked by d'Aquin *et al.* (2009), the process of extracting ontology modules should be guided by each domain or application. In this section we present our definition of ontology modules, and the methodology followed to obtain them.

3.1 Balanced SNOMED CT Modules

As input, we used the OWL-EL version of SNOMED CT obtained using the Perl script included in the distribution as input (SCT). For our purposes, presented in the introduction, we define a *balanced SNOMED CT module* (M) as a minimal collection of classes from SCT that conform to the following requirements:

(a) All classes in M are hierarchically connected to SNOMED CT's root concept in the same way as in SCT.
(b) All classes in M share the same axiomatical class definition as in SCT.
(c) Sub-hierarchies in M are distributed (approximately) in the same proportion as in SCT. In practical terms, when visualized using a treemap, M should look similar to the treemap of SNOMED CT shown in Figure 1.
(d) Our model is restricted to classes. SNOMED CT metadata concepts are not subject to modularization.

3.2 Module Construction from Seeds

To create our module M, we followed a similar approach to Seidenberg and Rector (2006). Using their terminology, concepts (in our case, classes) are represented as nodes in a graph, and seed concepts are called *target nodes*. The strategy consists in iteratively adding classes appearing in the right-hand expressions of their definitions, starting from seeds in a initial signature. Figure 2 shows an example of a resulting module, where it can be seen that (a) all classes are hierarchically connected to the root concept in the same way as in the original ontology (Figure 3), and (b) all classes share the same axiomatical class definition as the original ontology.

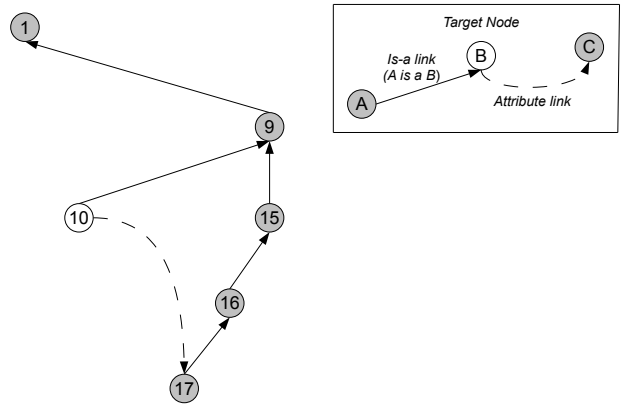


Fig. 2: Strategy followed to build our module M, starting from the seed concept (target node) 10. Figure 3 shows the original ontology from which it was extracted.

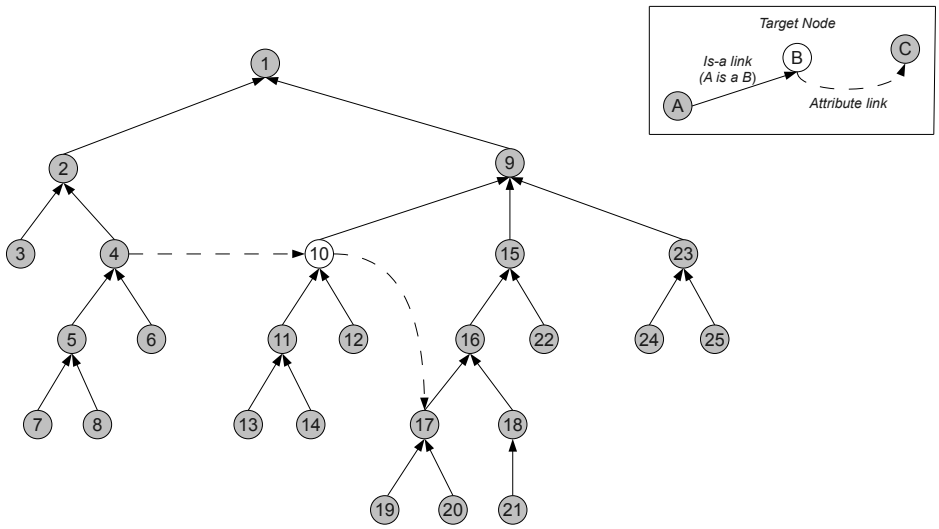


Fig. 3: Sample ontology, starting with a signature containing the seed node (target node) 10.

3.3 Seed Adjustment: An Iterative Algorithm

The strategy to build a module using seeds presented in the previous section guarantees requirements (a) and (b) from our definition of M, but does not guarantee requirement (c), i.e., that sub-hierarchies in M will be distributed (approximately) in the same proportion as in SCT. The reason is that there is no control over classes from other sub-hierarchies that are added in the process when following the right-hand expressions of the seeds.

Therefore, in order not to conflict with requirements (a) and (b) when creating M, the only possibility is to carefully select the initial signature that bootstraps the modularization algorithm. For that purpose, we investigated an iterative algorithm that dynamically adjusts the distribution of classes used as seeds in the initial signature. Before presenting the algorithm, we introduce the following notation:

- As introduced before, SCT represents the OWL EL version of SNOMED CT used as input. Sub-hierarchies are termed SH_k.
- M represents, the output module, whose sub-hierarchy distribution (Table 1) should match SCT's as much as possible.
- $SIGN$, is the input signature, consisting of classes from SCT, that is used to boostrap the modularization process described in Subsection 3.2.
- $Error(SH_k) = Size(M_{SH_k}) - Size(SCT_{SH_k})$ indicates the error on a per sub-hierarchy basis. Errors are calculated in percentage terms (see distribution in Table 1).
- $RSS = \frac{1}{18}\sum_{k=1}^{18} Error(SH_k)^2$, where RSS represents the residual sum of squares. Convergence of the algorithm is defined when $RSS < 1$.

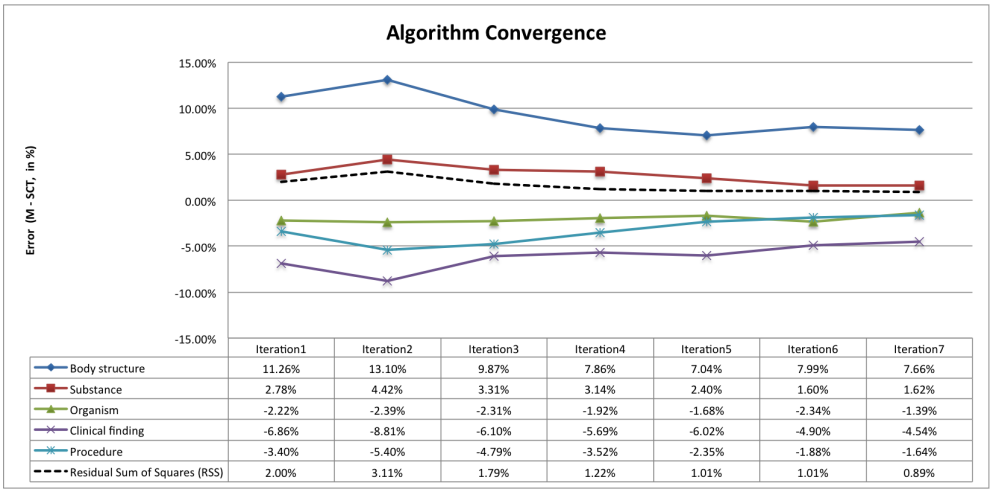
The algorithm, at each iteration i is the following:

1. A random signature $SIGN_i$ consisting of 2000 classes from SCT is selected, following the same class sub-hierarchy distribution as SCT, and ensuring at all sub-hierarchies in the signature contains at least one class.
2. A module M_i is computed following the principles described in Subsection 3.2. Its sub-hierarchy distribution is calculated.
3. Convergence is checked. If $RSS >= 1$, Steps 1 to 3 are repeated after adjusting the scaling factor for the sub-hierarchy distribution of the signatures in the next iteration $i+1$: $f(SIGN_{i+1_{SH_k}}) = f(SIGN_{i_{SH_k}}) \times \frac{f(SCT_{SH_k})}{f(M_{i_{SH_k}})}$ with $f(M_{i_{SH_k}})$ being the relative frequency of sub-hierarchy SH_k measured in the resulting module in iteration i, M_i.

4 RESULTS

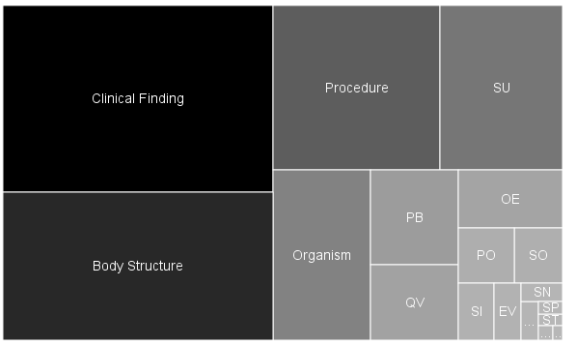
In our experiments, the algorithm converged after 7 iterations, extracting a module M with 10 834 classes. Figure 4 (Page 4) shows the error after each iteration for sub-hierarchies with more than 1% error, as well as the residual sum of squares.

As can be seen in the table below the graph, the sub-hierarchies *Clinical Finding*, *Procedure*, and *Organism* were under-represented in M, while *Body Structure* and *Substance* were over-represented. The same results can be confirmed graphically in the treemaps shown in Figure 5, at iterations 1, 3, and 7.

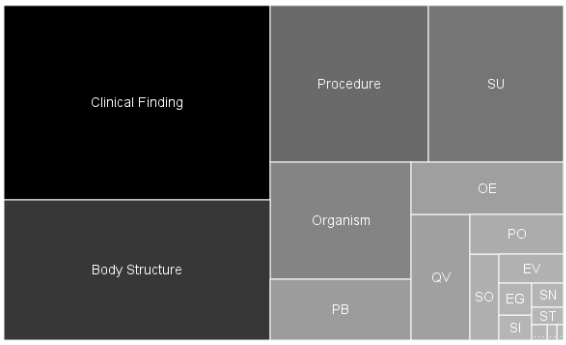


	Iteration1	Iteration2	Iteration3	Iteration4	Iteration5	Iteration6	Iteration7
Body structure	11.26%	13.10%	9.87%	7.86%	7.04%	7.99%	7.66%
Substance	2.78%	4.42%	3.31%	3.14%	2.40%	1.60%	1.62%
Organism	-2.22%	-2.39%	-2.31%	-1.92%	-1.68%	-2.34%	-1.39%
Clinical finding	-6.86%	-8.81%	-6.10%	-5.69%	-6.02%	-4.90%	-4.54%
Procedure	-3.40%	-5.40%	-4.79%	-3.52%	-2.35%	-1.88%	-1.64%
Residual Sum of Squares (RSS)	2.00%	3.11%	1.79%	1.22%	1.01%	1.01%	0.89%

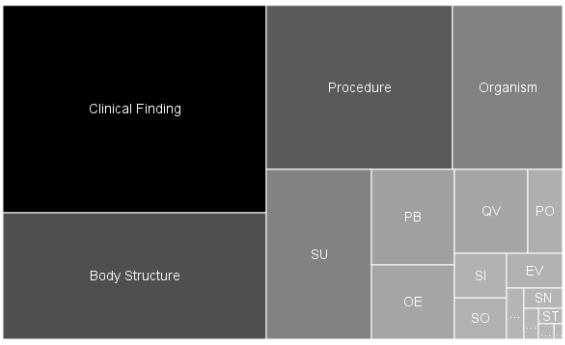
Fig. 4: Execution of the algorithm, showing convergence in iteration 7.



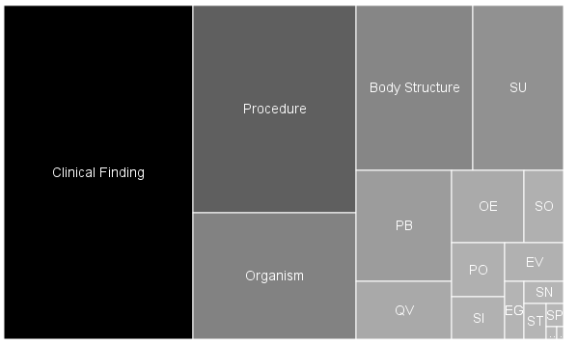
(a) Module Shape - Iteration 1



(b) Module Shape - Iteration 3



(c) Module Shape - Iteration 7 (convergence)



(d) Full SNOMED CT Shape (target)

Fig. 5: Visual comparison of the shape between the modules and SNOMED CT (d) in iterations 1 (a), 3 (b), and 7 (convergence, c). Clinical Finding, Procedure, and Organism were under-represented, while Body Structure and Substance were over-represented.

5 DISCUSSION

Our results suggest that it is difficult for ontology modules to meet all of our modularization criteria without relaxing the constraints of how concepts in the modules are distributed by sub-hierarchies, because modularization criteria are conflicting. In our experiments, all obtained modules over-represented or under-represented some of SNOMED CT's sub-hierarchies in different degrees. These results were partly expected, due to the nature of the modularization approach that uncontrollably adds class definitions to preserve SNOMED CT's hierarchy and class definitions.

The error figures that we obtained after convergence, however, never reached 8% for any sub-hierarchy and all our modules contained a fair representation of all of them. Furthermore, convergence was reached after only 7 iterations. Such modules might be sufficient in many of the use cases that motivated their creation, i.e., extracting modules that show an (approximately) concept distribution to the one shown in SNOMED CT.

6 CONCLUSIONS AND FUTURE WORK

In this study, we have studied SNOMED CT sub-hierarchies and proposed and evaluated an iterative algorithm for extracting compact modules that preserve the shape of SNOMED CT that we termed *balanced modules*. Extracting such modules has generally been neglected by work on ontology modularization, even though there are many use cases where balanced modules constitute an extremely valuable tool, such as in ontology-based quality assurance, scaling experiments for real-time performance, or developing scalable testbeds for software tools. Our proposed algorithm and our resulting modules show that graph-traversal ontology modularization techniques can effectively be used to create balanced modules, if the concept distribution of the input signature is dynamically and iteratively adjusted.

It is important to note that our algorithm and experiments are still at an initial stage and some aspects need to be further explored and more carefully evaluated. As future work, we plan to further (a) analyze how to select a minimal signature, (b) study how signature size influences the final size of the modules, and (c) improve the randomization process of the signature selection, e.g., by stratifying the randomization by node depth.

Our current results, however, show that SNOMED CT can indeed be squeezed without losing its shape, provided that we accept a moderate (up to 8%) under- and over-representation of some of its hierarchies.

ACKNOWLEDGMENTS

The authors acknowledge ICBO reviewers for their elaborate feedback and suggestions.

REFERENCES

Agrawal, A., Perl, Y., and Elhanan, G. (2012). Identifying problematic concepts in snomed ct using a lexical approach. *Studies in health technology and informatics*, **192**, 773–777.

Cuenca Grau, B., Horrocks, I., Kazakov, Y., and Sattler, U. (2008). Modular reuse of ontologies: Theory and practice. *Journal of Artificial Intelligence Research*, pages 273–318.

Doran, P., Tamma, V., and Iannone, L. (2007). Ontology module extraction for ontology reuse: an ontology engineering perspective. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 61–70. ACM.

d'Aquin, M., Schlicht, A., Stuckenschmidt, H., and Sabou, M. (2007). Ontology modularization for knowledge selection: Experiments and evaluations. In *Database and Expert Systems Applications*, pages 874–883. Springer.

d'Aquin, M., Schlicht, A., Stuckenschmidt, H., and Sabou, M. (2009). Criteria and evaluation for ontology modularization techniques. In *Modular ontologies*, pages 67–89. Springer.

Fung, K. W., McDonald, C., and Srinivasan, S. (2010). The UMLS-CORE project: a study of the problem list terminologies used in large healthcare institutions. *Journal of the American Medical Informatics Association*, **17**(6), 675–680.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., and Schneider, L. (2002). Sweetening ontologies with dolce. In *Knowledge engineering and knowledge management: Ontologies and the semantic Web*, pages 166–181. Springer.

Grau, B. C., Horrocks, I., Kazakov, Y., and Sattler, U. (2009). Extracting modules from ontologies: A logic-based approach. In *Modular Ontologies*, pages 159–186. Springer.

López-García, P., Boeker, M., Illarramendi, A., and Schulz, S. (2012). Usability-driven pruning of large ontologies: the case of snomed ct. *Journal of the American Medical Informatics Association*, pages amiajnl–2011.

Noy, N. F. and Musen, M. A. (2004). Specifying ontology views by traversal. In *The Semantic Web–ISWC 2004*, pages 713–725. Springer.

Pathak, J., Johnson, T. M., and Chute, C. G. (2009). Survey of modular ontology techniques and their applications in the biomedical domain. *Integrated computer-aided engineering*, **16**(3), 225–242.

Rogers, J. and Rector, A. (1996). The galen ontology. *Medical Informatics Europe (MIE 96)*, pages 174–178.

Schulz, S. and Boeker, M. (2013). Biotoplite: An upper level ontology for the life sciencesevolution, design and application. In *GI-Jahrestagung*, pages 1889–1899.

Seidenberg, J. and Rector, A. (2006). Web ontology segmentation: analysis, classification and use. In *Proceedings of the 15th international conference on World Wide Web*, pages 13–22. ACM.

Smith, B., Kumar, A., and Bittner, T. (2005). Basic formal ontology for bioinformatics. *Journal of Information Systems*, pages 1–16.

Stuckenschmidt, H., Parent, C., and Spaccapietra, S. (2009). *Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization*. Springer-Verlag.

Wennerberg, P., Schulz, K., and Buitelaar, P. (2011). Ontology modularization to improve semantic medical image annotation. *Journal of biomedical informatics*, **44**(1), 155–162.

BIM: An Open Ontology for the Annotation of Biomedical Images

*Ahmad C. Bukhari[1], Mate Levente Nagy[2], Michael Krauthammer[2], Paolo Ciccarese[3], *Christopher J. O. Baker[1]

[1] Dept. Computer Science & Applied Statistics, University of New Brunswick, Saint John, NB, E2L 4L5, Canada

[2] Dept. Pathology & Yale Center for Medical Informatics, 300 Cedar Street, New Haven, CT 06510, USA

[3] Harvard Medical School, 25 Shattuck Street, Boston, MA 02115 USA

ABSTRACT

Biomedical images published within the scientific literature play a central role in reporting and facilitating life science discoveries. Existing ontologies and vocabularies describing biomedical images, particularly sequence images, do not provide sufficient semantic representation for image annotations generated automatically and/or semi-automatically. We present an open ontology for the annotation of biomedical images (BIM) scripted in OWL/RDF. The BIM ontology provides semantic vocabularies to describe the manually curated image annotations as well as annotations generated by online bioinformatics services using content extracted from images by the Semantic Enrichment of Biomedical Images (SEBI) system. The BIM ontology is represented in three parts; (i) image vocabularies - which holds vocabularies for the annotation of an image and/or region of interests (ROI) inside an image, as well as vocabularies to represent the pre and post processing states of an image, (ii) text entities - covers annotations from the text that are associated with an image (e.g. image captions) and provides semantic representation for NLP algorithm outputs, (iii) a provenance model - that contributes towards the maintenance of annotation versioning. To illustrate the BIM ontology's utility, we provide three annotation cases generated by BIM in conjunction with the SEBI image annotation engine.

1 INTRODUCTION

Images depicting key findings of research papers contain rich sets of information derived from a wide range of biomedical experiments. Biomedical imaging [1] employs numerous modalities such as X-Rays (CT scans), sound (ultrasound), magnetism (MRI), radioactive pharmaceuticals (nuclear medicine: SPECT, PET) or light (endoscopy, OCT) to evaluate the status of an organ or tissue. Unlike text or other non-imaging data, image data poses a number of idiosyncratic issues rendering them mainly opaque for reuse without significant manual intervention. Current practices related to the extraction of implicit knowledge provide annotations that are neither anchored with an image, nor documented in a machine-readable fashion. As a consequence images cannot be readily discovered or categorized based on their contents. In the case of biomedical images that contain some type of biological sequence data summarizing the atomic composition of biological molecules [2] a combination of optical character recognition and text extraction techniques can provide better searchability over these images such that questions like "*display of all the sequence images that show proteins from the same protein family*"- [3] could be asked, provided that annotations could be made available to a search or query engine. However, image repositories in use today restrict the features that users can search with to those described in text based image captions and predominantly encourage the syntactic keyword based search, which constitutes a significant limitation [4]. In contrast images with semantic annotation can be automatically and/or semi-automatically discovered and linked to new information. The resulting enriched images are readily reusable based on their semantic annotations and can be used in semantic search and ad-hoc data integration activities. Overall, to achieve a greater degree of reusability and interoperability over image data certain core infrastructure is required, including automated image annotation pipelines and semantic vocabularies that can anticipate and represent image related content unambiguously. Existing ontologies and vocabularies describing biomedical images, particularly sequence images, are not sufficient to fulfill the requirements mentioned above and for our use case (SEBI) [4]. This motivated us to build the BIM ontology described in this paper which was designed and modeled with the following purposes in mind: formal representation of image annotation, enhanced reusability of image related data, depiction of pre and post image processing phases, design of context aware image search engines and semantics enabled bioimaging applications.

2 THE BIM ONTOLOGY

To better understand the context where BIM is relevant we briefly describe SEBI (semantic enrichment of biomedical images). SEBI is a solution for image annotation

* To whom correspondence should be addressed: bakerc@unb.ca

that adopts a combination of technologies to comprehensively capture information associated with, and contained in, biomedical images. To achieve this SEBI utilizes information extracted from images as seed data to aggregate and harvest new annotations from heterogeneous online biomedical resources. SEBI incorporates a variety of knowledge infrastructure components and services including image feature extraction [5], semantic web data services [6], linked open data [7] and crowd annotation [8]. Together these resources make it possible to automatically and/or semi-automatically discover and semantically interlink the new information in a way that supports semantic search for images. The resulting enriched images are readily reusable based on their semantic annotations and can be used in ad-hoc data integration activities. To date the BIM ontology has been used to successfully annotate 15000 images from the Yale Image Finder [3], 85% automatically and 15% through manual crowdsourcing.

3 MATERIALS AND METHODS

BIM ontology has been created to provide the standardized semantic representation of the annotations generated to describe a biomedical image by SEBI. BIM can further be used for annotating the associated text references by a machine or human. In order to collect the relevant terms, relationships / properties for sequence related images, we reviewed literature mentioning sequence analysis algorithms [9] such as BLAST, HMMER, Prosite, and the conserved domain database. A total number of 23 papers published from 2006 to 2015 were selected from different journals. We focused on actual depictions and discussion of sequence alignment outputs, rather than the algorithms, to distill the typical terms, concept and relations used. In order to accumulate terminologies associated with non-sequence image types such as: X-Rays, ultrasound, MRI, radioactive pharmaceuticals endoscopy, we selected a random sample set of papers from the Journal of *Bioimaging* and applied the *SNOMED-CT*[1] *and BioNLP web services* [10] to expedite the knowledge elicitation process. The *SNOMED-CT* and NLP web services provided the exact annotation location (e.g. start and stop annotation word) wherever a term existed in the paper. Manual evaluation of the outputs extracted from papers was performed, whenever relevant terms were found they were categorized and documented. While modeling the BIM ontology, a number of ontologies relating to annotation and biomedical imaging were also consulted and where appropriate, classes and properties were reused.

Table 1 depicts the ontologies, prefixes and namespaces of the existing ontologies that have been employed in the modeling of BIM ontology. We have reused the vocabularies defined in Annotation Ontology (AO) [11] to model the biological concepts mentioned in an image caption. AO is an open-source ontology for annotating the scientific documents on the web. In AO, all the annotations are regarded as resources and fall under the instance category of the *Annotation class*. Each annotation has some *hasTopic*, context predicates and object class. Objects can be a particular entity such as protein or chemical name, a disease, or reified fact, while the context refers to a certain text segment inside the sentence (see Fig.3). This simple reference model makes it possible to integrate the extracted information semantically. The provenance of annotations is modeled with Provenance, Authoring and Versioning (PAV) ontology [12] e.g. predicates such as *createdBy*, *createdOn* describe the annotation creator and date of creation. PAV provides the terminologies for tracing provenance of the digital entities that have been published on the web and then accessed, transformed and consumed. To cover high-level scientific research concepts, terms from the *Semanticscience Integrated Ontology* (SIO) were imported [13]. SIO provides a simple, integrated ontology of types and relations to describe objects, processes and their attributes. SIO behaves as an upper level ontology and supplies many high-level biomedical concepts. To represent the structural information of a biological sequence semantically, we incorporated a number of classes and relationships from Sequence Ontology (SO) [14] ontology such as *transcript, primary-transcript, intron*, *mRNA, insertion sequence.*

Table 1. Well-known vocabularies utilized in BIM modeling

Ontology/Vocabulary	*Prefix*	*Namespace*
Annotation Ontology	AO	http://purl.org/ao/
Provenance Authoring & Versioning Ontology	PAV	http://purl.org/pav/
SemanticScience Integrated Ontology	SIO	http://semanticscience.org/ontology/sio.owl
Sequence Ontology	SO	http://purl.obolibrary.org/obo/so.owl
Friend Of A Friend	FOAF	http://xmlns.com/foaf/0.1/
SIOC Ontology	SIOC	http://rdfs.org/sioc/ns#
SKOS ontology	SKOS	http://www.w3.org/2004/02/skos/core
Exif Ontology	exif	http://www.kanzaki.com/ns/exif#
Time Ontology	TIME	http://www.w3.org/TR/owl-time/
Semantic DICOM Ontology	DICOM	http://purl.bioontology.org/ontology/SEDI
DBpedia Ontology	DBpedia	http://dbpedia.org/ontology/

The Exif[2] ontology [15] mainly describes the Exif format of picture data semantically, and provides useful vocabularies supporting the pre-processing and usage of Exif images. In BIM ontology, we used the Exif terminologies to define image orientation and size using *Ex-*

if:Orientation, Exif:ImageWidth, Exif:ImageHeight and corresponding vocabularies to represent the stages of image processing e.g *Exif:WhiteBalance.* DICOM (Digital Imaging and Communications in Medicine) [16] is a standard to represent the medical image information worldwide. Most of the available medical images modalities follow the DICOM standards to capture, store and disseminate the medical image information. However, the DO (DICOM Ontology) [17] serves the purpose of integrating and explicitly representing the concepts and relationships of DICOM in machine readable and human understandable format. In BIM ontology, we imported DO classes to represent the information associated with radiology images and to represent image capturing detail semantically. The FOAF [18] vocabulary describes people, their relations with other people, and objects that are related to a person-to-person connection.

We also leveraged the *DBpedia* ontology [19], a multi-domain ontology that is mainly designed to cover the Wikipedia infoboxes. In version 3.2, there are roughly 359 classes and 1775 properties, which cover a vast range of common and life science concepts. In contrast, the Dublin core Metadata [20] vocabulary was used to represent general meta-data attributes for documents such as titles, authors, subjects, descriptions, date, type, and format. Core concepts from time and relationship ontologies were imported to describe concepts relating to time units (e.g. minutes, seconds) and relations between objects. The Semantically-Interlinked Online Communities (SIOC pronounced as "shock") [21] is a domain ontology, which perfectly defines and interlinks all the online communities' concepts such as posts, comments, and users. Similarly, the Simple Knowledge Organization System (SKOS) [22] is a generalized model written in RDF for sharing and interlinking organizational knowledge with semantic description. We reused the terms *SKOS:prefLabel, SKOS:Concept, SIOC:Item and SIOC:userAccount* from SIOC and SKOS ontologies. To assemble the BIM ontology model, we used the *Protégé*, editor [23]. However, to efficiently manage and utilize the BIM vocabularies, an ontology-publishing server called *UNBvps (http://cbakerlab.unbsj.ca/unbvps/)* was set up.

The server provided a range of control functions, including management of provenance, versioning of the source vocabularies, and delete/update functions. We enhanced the Neologism plugin [24] on our server to reduce the time spent developing and publishing vocabularies with conventional ontology authoring techniques i.e. using *Protege* and internet publishing. To identify the appropriate semantic mappings between existing ontologies and BIM ontology, a Java program that suggests the possible mappings was created. The program extracted the tables and column names, storing them as variables and invoked a *WordNet*[3] web service that lexically compared each variable with the ontology entities to find possible matches. The overall goal was to provide candidate matches for subsequent curation; a comprehensive benchmarking of the algorithm's performance was not derived. A cursory evaluation of the derived mappings showed three types of results; (i) mappings that fully met our requirements, which suggested predicates such as *hasPubMedID* and *hasPMCID* in the *FRBR-aligned Bibliographic Ontology* [25] *(FaBio)*; (ii) mappings that were insufficiently defined, like the image Feature property that existed in BioPortal; and (iii) mappings with hosted resources that did not appear trustworthy.

4. USE CASES

This section demonstrates the BIM ontology modeling with three different use cases.

4.1 Use case 1: Automatic sequence image annotation

To perform enrichment of a biological sequence image with semantic annotations, a cluster of SADI web services [26] was developed. When the SEBI platform sends a request to semantically annotate an image, a number of web services are invoked serially. The image extraction and analysis service takes the image and applies the image processing filters to improve the image contrast and to improve the image resolution. Subsequently, the OCR extraction web service receives a processed image and applies an algorithm to extract the optical characters from the image. BIM ontology supplies the necessary vocabularies to express the pre and post image processing stages such as: *BIM:hasImageResolution* and *BIM:ImageFilters* used to semantically represent features that have been used to process an image. Subsequently the OCR extraction web service pulls out the sequence (optical characters) from an image while BIM ontology represents that sequence string as *BIM: SequenceBlock.* Later the extracted sequence string has been passed to the sequence analysis web services to generate annotations on a sequence image. The SADI sequence analysis service module has been designed to retrieve annotations for biological sequences from various biological sequence analysis tools such as *HMMER, BLAST, Pfam, ProSite, and GO*. Fig. 1 displays the semantic modeling provided by the BIM ontology to enrich a sequence image with semantic annotations. The annotations harvested by the sequence analysis services (by exposing sequence analysis software as web services) provide useful information about a sequence image. The newly generated annotation further underpins the image similarity module of SEBI that accurately fetches the relevant/similar sequence images from the scientific literature. To preserve the provenance of an image and annotations curated on an image, BIM ontology reuses the vocabularies provided by the PAV ontology as displayed in Fig.1. The terms such as: *pav:createdBy* and *pav:createdOn* have been recruited to represent the web service and the annotation creation date respectively. However, the terms such as

BIM:hasSequenceType, BIM:hasMutationResidue, BIM:hasConservedResidue, BIM:hasMOTIF, BIM:hasProteinInteractionSite explicitly define the outputs of sequence analysis software. All terms relating to sequence analysis have been defined for the first-time in BIM ontology, as we did not find their accurate representation in any ontologies available online. Additionally, we can utilize *time:Instant* to capture the hours, minutes and seconds for *createdOn*.

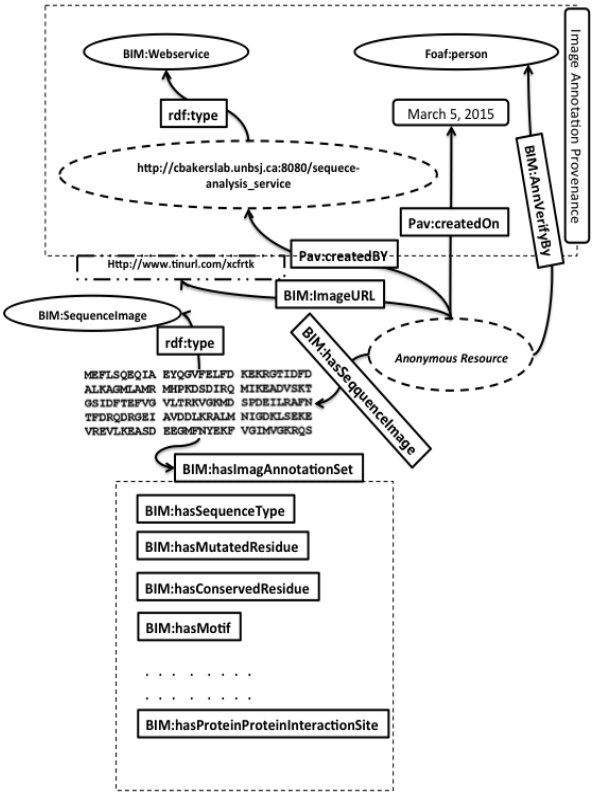


Fig. 1 BIM Model of Automatic Sequence Annotation by a Web Service

4.2 Use Case 2: Crowd-based semi automatic annotation

Semi-automatic annotation, where automatic annotation is not feasible due to poor quality input images, is made possible through the introduction of a crowd annotation technique. All images that fail to produce new annotations through web services are forwarded to the crowd annotation module of SEBI. Salient features of the crowd annotation module are as follows: Users can annotate, delete, or update annotations, maintain private annotations or share them with other legitimate users. BIM provides vocabularies through which a user can maintain image provenance, for instance it documents the author (human or machine) that has curated an annotation and the location *(xy-coordinates)* inside an image. Moreover, the crowd annotation module provides a utility through which a user can select and annotate a portion within an image. To support such activities BIM ontology supplies the crowd annotation module with *BIM:CTScan*, *BIM: hasSomeLesion*, and *BIM: polygonCoordinates* to semantically express the intra image annotation and the position of the annotation inside an image. *BIM: Resolution* class has further subclasses in *BIM: Width sameAS Exif:ImageWidth* and *BIM: Height sameAS Exif:Imageheight.* The *BIM:AnnotationRevision* class facilitates a user to track the legacy annotation made on an image along with information on the creator/software agent.

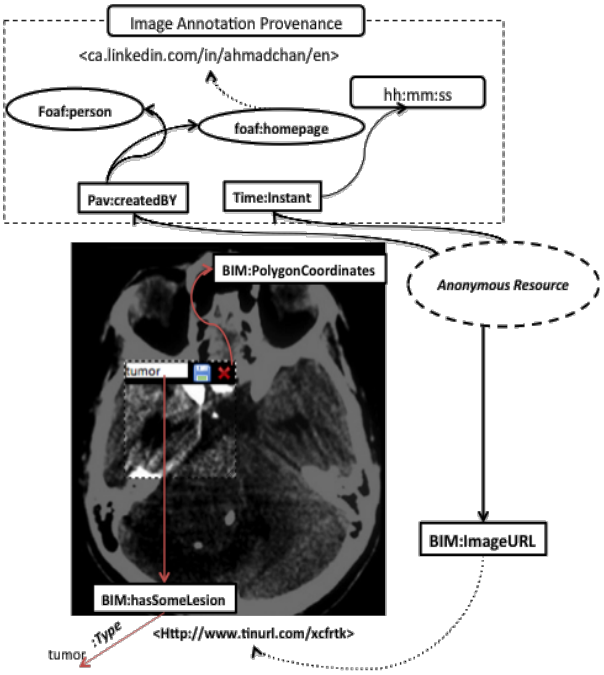


Fig. 2 BIM Crowd-Sourced Modeling of a Biomedical Image

4.3 Use Case 3: Text associated with an image

In SEBI, the BioNLP annotation module extracts named entities, such as drug names, diseases, chemicals, proteins, lipids or GO terms found in the captions or in the descriptions of a biomedical image in a paper. The BioNLP annotation module further normalizes the entities to canonical names defined in online resources e.g. PDB and DrugBank and publishes them in RDF to annotate the images. The BIM ontology incorporates the Annotation Ontology and PAV ontology vocabularies to semantically annotate the concepts and relationships. Fig. 3 explains the BIM ontology modeling on the caption of an image where a drug is

[1]http://ihtsdo.org/snomed-ct/

[2]http://www.kanzaki.com/ns/exif

[3]http://wordnet.princeton.edu/

[4]http://www.rcsb.org/pdb/home/home.do

[5]http://www.drugbank.ca/

mentioned.

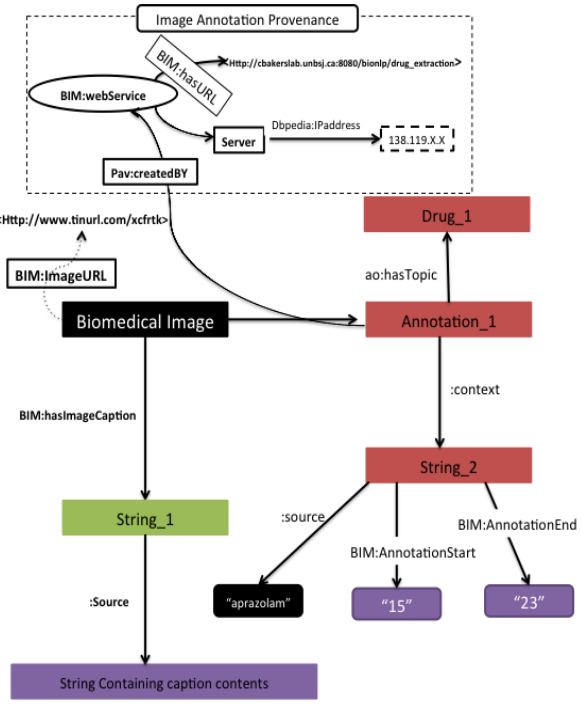


Fig. 3 BIM Ontology Modeling for Image Associated Text

CONCLUSIONS

This paper introduces Biomedical Image ontology (BIM) that supports the publication of annotations on, features identified within a biomedical image generated by SEBI tools. BIM was created to address the dearth of appropriate ontologies and appropriately integrated semantic metadata targeted to annotating diverse biomedical images, particularly images depicting biological sequences. BIM supports both the creation of machine generated and human curated annotations which can be reused in multiple knowledge discovery tasks or resources. These include; image mashups, linked image data, semantic image search and the computing of image similarity, which along with provenance annotations indicating an image's source publication permits the linking of publications containing related images. The SEBI framework is designed to facilitate all these goals.

REFERENCES

1. Perkel, J. M. (2013). *Life Science Technologies: Mass Spec Imaging: From Bench to Bedside*. Science, 340(6136), 1119-1121
2. Webb, A., & Kagadis, G. C. (2003). *Introduction to biomedical imaging*. Hoboken: Wiley.
3. Xu, S., McCusker, J., & Krauthammer, M. (2008). *Yale Image Finder (YIF): a new search engine for retrieving biomedical images.* Bioinformatics, 24(17), 1968-1970.
4. Bukhari, A. C., Krauthammer, M., & Baker, C. J. SEBI: *An Architecture for Biomedical Image Discovery, Interoperability and Reusability based on Semantic Enrichment.*
5. Corke, P. (2011). *Image Feature Extraction*. In Robotics, Vision and Control (pp. 335-379). Springer Berlin Heidelberg.
6. Fensel, *D., Facca, F. M., Simperl, E., & Toma, I. (2011). Semantic web services. Springer* Science & Business Media.
7. Bizer, C., Heath, T., & Berners-Lee, T. (2009). *Linked data-the story so far.*
8. Dijkshoorn, C., Oosterman, J., Aroyo, L., & Houben, G. J. (2012, July). *Personalization in crowd-driven annotation for cultural heritage collections.* In4th International Workshop on Personalized Access to Cultural Heritage PATCH 2012, Montreal, Canada, July 16-20, 2012.
9. Li, H., & Homer, N. (2010). *A survey of sequence alignment algorithms for next-generation sequencing.* Briefings in bioinformatics, 11(5), 473-483.
10. Bukhari, A. C., Klein, A., & Baker, C. J. (2013, January). *Towards Interoperable BioNLP Semantic Web Services Using the SADI Framework.* In Data Integration in the Life Sciences (pp. 69-80). Springer Berlin Heidelberg.
11. Ciccarese, P., Ocana, M., Garcia-Castro, L. J., Das, S., & Clark, T. (2011). *An open annotation ontology for science on web 3.0. J. Biomedical Semantics*,2(S-2), S4.
12. Ciccarese, P., Soiland-Reyes, S., Belhajjame, K., Gray, A. J., Goble, C. A., & Clark, T. (2013). *PAV ontology: provenance, authoring and versioning. J. Biomedical Semantics,* 4, 37.
13. Dumontier, M., Baker, C. J., Baran, J., Callahan, A., Chepelev, L. L., Cruz-Toledo, J., ... & Hoehndorf, R. (2014). *The Semanticscience Integrated Ontology (SIO) for biomedical research and knowledge discovery*. J. Biomedical Semantics, 5, 14.
14. Eilbeck, K., Lewis, S. E., Mungall, C. J., Yandell, M., Stein, L., Durbin, R., & Ashburner, M. (2005). *The Sequence Ontology: a tool for the unification of genome annotations*. Genome biology, 6(5), R44.
15. Alvarez, P. (2004). *Using extended file information (EXIF) file headers in digital evidence analysis*. International Journal of Digital Evidence, 2(3), 1-5.
16. Mildenberger, P., Eichelberg, M., & Martin, E. (2002). *Introduction to the DICOM standard. European radiology*, 12(4), 920-927.
17. Kahn Jr, C. E., Langlotz, C. P., Channin, D. S., & Rubin, D. L. (2011). *Informatics in Radiology: An Information Model of the DICOM Standard 1.*Radiographics, 31(1), 295-304.
18. Golbeck, J., & Rothstein, M. (2008, July). *Linking Social Networks on the Web with FOAF: A Semantic Web Case Study*. In AAAI (Vol. 8, pp. 1138-1143).
19. Töpper, G., Knuth, M., & Sack, H. (2012, September). *Dbpedia ontology enrichment for inconsistency detection.* In Proceedings of the 8th International Conference on Semantic Systems (pp. 33-40). ACM.
20. Dublin Core Metadata Initiative. (2012). Dublin core metadata element set, version 1.1.
21. Breslin, J. G., Decker, S., Harth, A., & Bojars, U. (2006). *SIOC: an approach to connect web-based communities*. International Journal of Web Based Communities, 2(2), 133-142.
22. Miles, A., & Pérez-Agüera, J. R. (2007). *Skos: Simple knowledge organisation for the web. Cataloging & Classification Quarterly*, 43(3-4), 69-83.
23. Rubin, D. L., Noy, N. F., & Musen, M. A. (2007). *Protege: a tool for managing and using terminology in radiology applications*. Journal of Digital Imaging, 20(1), 34-46.
24. Basca, Cosmin et al. (2008). Neologism: Easy Vocabulary Publishing.
25. Shotton, D., & Peroni, S. (2011). *FaBiO: FRBR Aligned Bibliographic Ontology*.
26. Wilkinson, M. D., Vandervalk, B. P., & McCarthy, E. L. (2011). *The Semantic Automated Discovery and Integration (SADI) Web service Design-Pattern, API and Reference Implementation*. J. Biomedical Semantics, 2(8).

Availability: The BIM ontology, version 1.0 is scripted in OWL/RDF is available on Bioportal and can be accessed at: http://bioportal.bioontology.org/ontologies/BIM

Investigating Term Reuse and Overlap in Biomedical Ontologies

Maulik R. Kamdar, Tania Tudorache, and Mark A. Musen
Stanford Center for Biomedical Informatics Research, Department of Medicine, Stanford University

ABSTRACT

We investigate the current extent of term reuse and overlap among biomedical ontologies. We use the corpus of biomedical ontologies stored in the BioPortal repository, and analyze three types of reuse constructs: (a) explicit term reuse, (b) *xref* reuse, and (c) Concept Unique Identifier (CUI) reuse. While there is a term label similarity of approximately 14.4% of the total terms, we observed that most ontologies reuse considerably fewer than 5% of their terms from a concise set of a few core ontologies. We developed an interactive visualization to explore reuse dependencies among biomedical ontologies. Moreover, we identified a set of patterns that indicate ontology developers did intend to reuse terms from other ontologies, but they were using different and sometimes incorrect representations. Our results suggest the value of semi-automated tools that augment term reuse in the ontology engineering process through personalized recommendations.

1 INTRODUCTION

Ontologies have been used in biomedical research for different purposes: knowledge management, semantic search, data annotation, data integration, exchange, decision support and reasoning (Bodenreider, 2008; Rubin *et al.*, 2008). Biomedical ontologies range drastically in their size, coverage of a domain, and in their level of adoption. It is only natural that biomedical ontologies will overlap to a certain degree, as they sometimes need to represent common parts of a domain, or different domains that have shared terms.

Several large biomedical efforts deal in different ways with managing the overlap of ontologies and reuse. For example, the Open Biological and Biomedical Ontologies (OBO) Foundry (Smith *et al.*, 2007) aims to create a set of *"orthogonal"* ontologies, such that each term is defined only in one ontology, and is referred using its Internationalised Resource Identifier (IRI) in other ontologies. The OBO ontologies use the *xref* mechanism to create references between terms in different ontologies (OBOFoundry, 2011). To support the interoperability across different biomedical ontologies and terminologies, the Unified Medical Language System–UMLS (Bodenreider, 2004) uses the notion of a Concept Unique Identifier (CUI) to map terms with similar meaning in different terminologies.

All ontology development methodologies strongly encourage reuse while building new ontologies, be it at the level of an ontology, or at the level of the terms (Corcho *et al.*, 2003; Alexander, 2006). Reuse has some directly apparent advantages, such as, developing a unified theory of biomedicine, semantic interoperability and reducing engineering costs (since reuse avoids rebuilding existing ontology structures). For example, the 11th revision of the International Classification of Diseases (ICD-11) reuses terms from other established ontologies, such as SNOMED CT, to spare the effort in creating already existing quality content (e.g., the anatomy taxonomy), to increase their interoperability, and to support its use in electronic health records (Tudorache *et al.*, 2010). Another benefit of the ontology term reuse is that it enables federated search engines to query multiple, heterogeneous knowledge sources, structured using these ontologies, and eliminates the need for extensive ontology alignment efforts (Kamdar *et al.*, 2014).

For the purpose of this work, we define as **term reuse** the situation in which the same term is present in two or more ontologies either by direct use of the same identifier, or via explicit references and mappings. We define as **term overlap** the situation in which the term labels in two or more ontologies are lexically similar (see Section 3). We further classify the reuse: (1) *Reuse of an ontology*, through the means of the import mechanism available in OWL (W3C, 2012), meaning that the entire source ontology is imported into the target ontology; and (2) *Reuse of terms* from one source ontology into another. In many cases, experts reuse not only one term from one ontology, but rather subsets of terms from multiple ontologies (e.g., subtrees).

The goal of this work is to investigate the level of reuse and ovelap among biomedical ontologies. We harvested the ontologies from BioPortal (Whetzel *et al.*, 2011), an open content repository of biomedical ontologies and terminologies, and ran several analyses that show not only the level of reuse, but also how the reuse occurs.

The contributions of this work are threefold:

1. A set of descriptive statistics for the level of reuse in biomedical ontologies,
2. An interactive visualization technique for displaying the reuse dependencies among biomedical ontologies,
3. A discussion on the state and challenges of reuse in biomedical ontologies.

2 RELATED WORK

Through a set of use cases in bio-medicine and eRecruitment, Bontas *et al.* (2005) emphasise the need for more pragmatic methods and semi-automated tools that allow developers to exploit the vocabulary of domain-specific source ontologies for reuse. Matentzoglu *et al.* (2013) provide a method to analyze the overlap between ontologies in automatically-generated random snapshots of the OWL Web. Ontologies with 90% overlap or containment relations were considered similar. Ghazvinian *et al.* (2011) describe an approach to determine the level of orthogonality and *term overlap* (term label similarity) in OBO Foundry member and candidate ontologies. Their analysis over a period of two years indicated that, while the OBO Foundry has made significant progress towards achieving *"orthogonality"*, *term overlap* between

ontologies has remained consistent. Poveda Villalón *et al.* (2012) analyze the landscape of reuse in the ontologies used in Linked Open Data (LOD), and find that over 40% of the terms are reused from other vocabularies, 67% of which are reused by imports, and the rest by referencing the term IRI. Ontology modularisation techniques (i.e., extracting parts of an ontology using some structural or logical properties) are also an important factor in supporting reuse. Comprehensive studies of existing modularization techniques have also been undertaken (d'Aquin *et al.*, 2009; Pathak *et al.*, 2009).

There are only a few tools that support term reuse in biomedical ontologies. OntoFox (Xiang *et al.*, 2010) is a Web-based application that allows users to retrieve terms, selected properties, and annotations from the source ontologies, using MIREOT principles (Courtot *et al.*, 2011). The BioPortal Import Plugin (Nair, 2014), MIREOT Protégé Plugin (Hanna *et al.*, 2012) and DOG4DAG (Wächter *et al.*, 2011) are provided as extensions to the Protégé ontology editor (Noy *et al.*, 2001) to allow the import of terms, their properties, and even class subtrees from BioPortal.

3 METHODS

We obtained a triplestore dump of the BioPortal ontologies in N-triples format as of January 1, 2015, which contained 377 distinct biomedical ontologies. This dump does not contain some ontologies that were deprecated or merged with existing ontologies, or those that were added to BioPortal after January 1, 2015. These ontologies include eight OBO Foundry member ontologies (GO, CHEBI, PATO, OBI, ZFA, XAO, PR and PO) and 31 UMLS Terminologies (SNOMED CT, ICD-9, etc.). To conduct our analysis, we identified three constructs that cover reuse in BioPortal ontologies:

1. **Explicit reuse construct**: The IRIs of the terms in different ontologies are exactly the same.
2. **xref construct:** One term contains a reference to the other term IRI using the *xref* predicate.
3. **UMLS CUI construct:** Two BioPortal-defined term IRIs are mapped to the same Concept Unique Identifier.[1]

By iterating over all the asserted axioms in each of these 377 ontologies, we extracted all the class term IRIs, their labels, *xref* links and UMLS CUI mappings, when available. From the 5,718,276 class terms, we used the above three constructs to extract the set of terms that satisfy any of the three reuse criteria. The *xref* axioms were further filtered to separate those that assert equivalence between the connected entities (e.g., `CL:0000066`, `CARO:0000077` and `FMA:66768` all refer to *'epithelial cell'*), from those that were either references to resources in external databases like PubMed, or entities that were semantically treated as *genus-differentia* definitions, as defined in the OBOFoundry (2011).

For the first two reuse types (*Explicit* and *xref*),[2] we identified the source ontology for each term by converting each term IRI to lowercase and using *RegExp* filters constructed from ontology namespaces and common identifier patterns. A heuristic approach was used to determine the source ontology, by first checking only the current ontologies that share this term. We found some instances where the source is not determined in the first step. For example, `NCIT:Cerebral_Vein` is reused by Sage Bionetworks Synapse Ontology (SYN) and Cigarette Smoke Exposure Ontology (CSEO). However, this term is replaced by `NCIT:C53037` in the current version of NCI Thesaurus, and the original term is not present. Hence, as a second step, we extended our search to include all the ontologies. This two-step approach also deals with the conditions when an ontology acronym ('PR') is present in a term IRI (e.g., `Protein`) but is not necessarily in the source ontology.

We normalised the term labels by converting them to lowercase and removing all non-alphanumeric characters. We performed naïve string matching on the term labels to determine the potential *term overlap*.

We calculated three statistics:

1. The percentage of terms explicitly reused or *xref*-linked by an ontology, and the total number of ontologies these terms are reused from (on *Explicit* and *xref* constructs),
2. The percentage of terms and the total number of ontologies that are reused explicitly, or *xref*-linked, from an ontology (on *Explicit* and *xref* constructs).
3. The reuse between all distinct pairs of ontologies (on *Explicit*, *xref*, and *UMLS CUI* constructs).

Using these statistics, we determined which ontologies reused the maximum number terms from other ontologies, and also those ontologies whose terms were reused the most. We calculated the gap between term overlap and term reuse by subtracting the matched labels of reused terms.

To determine the level of import at the level of an ontology, we used the explicit occurrence of the `owl:imports` in the ontology files. However, this method did not account for the cases in which the imported ontologies were already merged into the importing ontology, as is the case in BioPortal. Therefore, we established an empirical threshold of 35% on the number of terms that were reused with respect to the total number of classes in the source ontology, above which we would consider the term reuse as a *reuse of the entire ontology*. As determined from reuse statistics, this threshold would allow us to consider term reuse from older versions of source ontology as ontology reuse.

During the development of an ontology for a specific domain, it is beneficial for the ontology engineers to have an idea regarding the set of ontologies whose terms were reused from by other related ontologies. Hence, we developed an interactive force-directed network visualization to represent the ontology pairs derived from the third statistic to explore the reuse dependencies among biomedical ontologies.

4 RESULTS

Explicit Reuse First, we investigated the **reuse at the level of an ontology** by the means of `owl:imports` mechanism and the 35% threshold method. The top 10 of the most imported ontologies are shown in Table 1.

[1] This was only checked for UMLS terminologies.

[2] UMLS CUI reuse was excluded, as we could not identify the source ontology for a CUI.

Ontology Name	#
(BFO) Basic Formal Ontology	59
(STY) Semantic Types Ontology	29
(PATO) Phenotypic Quality Ontology	10
(IAO) Information Artifact Ontology	9
(UO) Units of Measurement Ontology	5
(CARO) Common Anatomy Reference Ontology	4
(ONL-MSA) OntoNeuroLOG - Mental State Assessment	3
(ORDO) Orphanet Rare Disease Ontology	2
(GO) Gene Ontology	2
(BP) BioPAX Ontology of Biological Pathways	3

Table 1. Most imported ontologies (Reuse of an ontology). (#) indicates number of ontologies importing the specified ontology.

Second, we investigated the explicit **reuse of individual terms**. Of the 5,718,276 class terms that we extracted from the 377 BioPortal ontologies, we found 175,347 terms (3.1%) were explicitly shared among more than two ontologies using the same IRIs. We found the source ontology for all but 37 terms, which were primarily upper-level, abstract terms, whose ontologies were not present in BioPortal (e.g., `owl:Thing` and `time#datetimedescription`). After removing the terms that come from imported ontologies that were merged (term reuse > 35% threshold), we were left with only 59,618 terms (1.1%) actually reused.

xref Reuse We found a total of 4,370,350 *xref* axioms across all the BioPortal ontologies. After extracting *xrefs*, which assert equivalence between BioPortal ontology terms, we found 171,069 'outlinking' terms (3.9%) *xref*-linked to 386,442 'inlinking' terms (8.84%).

We also tried to understand how the explicit- and *xref* reuse is spread across different ontologies. **Figure 1** shows histograms of the percentage of terms reused by different ontologies. We can see that most ontologies reuse or *xref*-link less than 5% of their total terms. There were at least 150 ontologies which did not reuse a single term from other ontologies. We also observe that there are 20 ontologies that exhibit a reuse between 95% to 100% of their total terms. These ontologies are developed by reusing combinations of multiple ontologies (e.g., CCONT reuses terms from EFO, NCBITAXON, ORDO, and 19 other ontologies).

The top 10 ontologies that reuse their terms from the maximum number of ontologies, and those whose terms are reused the most, are shown in Tables 2 and 3, respectively. The columns indicate the percentage (%) of total terms explicitly reused or *xref*-linked from/by the number of ontologies (#). For example, current version of NIFSTD explicitly reuses 89.6% of its total terms from 42 different ontologies, and 95.2% and 3.7% of the total terms in the current version of GO are reused and *xref*-linked by 74 and 37 ontologies respectively. The top 10 terms, which are not upper ontology terms (e.g., from BFO or IAO) and are explicitly reused the most, are shown in Table 4.

UMLS CUI Reuse Using our third construct, we found 236,460 Concept Unique Identifiers (CUIs), which are mapped to more than two terms in UMLS terminologies. Some of the most mapped CUI terms are: `Neoplasms (C0027651)` and `Diabetes mellitus (C0011849)` appearing in 18 ontologies, `Schizophrenia (C0036341)` and `Leukemia (C0023418)` appearing in 17 ontologies. The full list is available online (see link at the end of section). **Figure 2** shows the percentage of CUI terms shared by each UMLS terminology with other terminologies. It is noteworthy to see some of the popular UMLS terminologies such as ICD10CM, LOINC, HL7 and MESH to be composed primarily of unshared, unique terms.

Ontology *(explicit)*	% Reused	# Reused	Ontology *(xref)*	% *xref*-linked	# *xref*-linked
NIFSTD	89.6	42	UBERON	72.2	37
HUPSON	55.8	32	CL	14.2	21
OBI_BCGO	97.9	25	TMO	17.3	21
IDOMAL	43.5	24	HPIO	53.7	16
IDODEN	29.1	23	DOID	90.8	13
OBI	19.1	22	TRAK	23.8	10
CCONT	98.8	22	GO	0.76	9
EFO	70.1	21	HP	11.8	8
CLO	7.2	19	DERMO	25.7	7
IDOBRU	43.27	19	EFO	0.76	6

Table 2. Ontologies that reuse the maximum number of terms from other ontologies - Percentage (%) of the total number of terms reused from the total number of ontologies (#).

Ontology *(explicit)*	% Reusing	# Reusing	Ontology *(xref)*	% *xref*-linking	# *xref*-linking
BFO	259	81	GO	3.7	24
GO	95.2	74	CHEBI	3.2	16
IAO	72.8	55	CARO	572	16
OBI	43.1	51	MESH	2.3	11
PATO	190	45	PATO	23.6	10
CHEBI	54.2	37	FMA	14.0	10
CL	15.4	36	NCIT	6.6	10
NCBITAXON	0.30	30	CL	18.9	9
STY	100	29	NCBITAXON	19.9	8
UO	136	27	SO	5.0	8

Table 3. Ontologies whose terms are reused most by other ontologies - Percentage (%) of the total number of terms in the current version reused by the total number of ontologies (#).

Term IRI	Term Label	#Reusing Ontologies
GO:0008150	biological_process	33
OBI:0000011	planned process	31
OBI:0100026	organism	28
CHEBI:23367	molecular entity	26
NCBITaxon:9606	Homo sapiens	26
PATO:0001241	physical object quality	24
GO:0005575	cellular_component	23
PATO:0001995	organismal quality	23
PATO:0000001	Quality	22
NCBITaxon:10239	Virus	20

Table 4. Top 10 terms that are reused by maximum ontologies

Overlap Executing normalised string matching on the term labels, we found a *term overlap* of 823,621 shared term labels (14.4%). On removing those terms that were already explicitly reused using the same term IRI, we reduced the list to 752,176 labels (13.2%). On removing those terms which were mapped to the same UMLS CUI, we further reduced the list to 617,509 labels (10.8%). On extracting

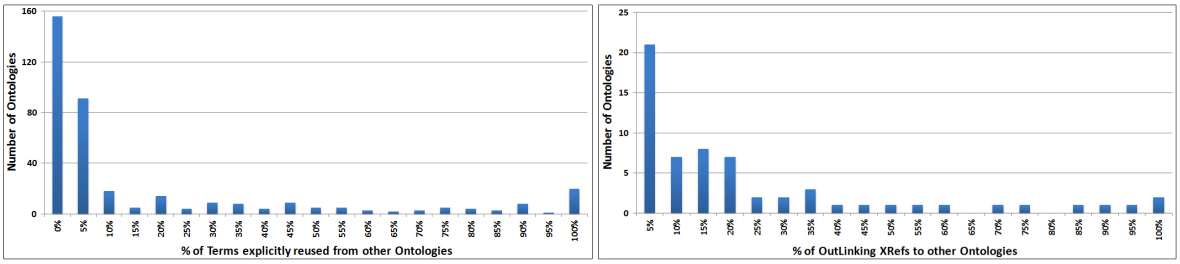


Fig. 1. Histograms depicting the first statistic: **a)** Percentage (%) of terms explicitly reused or **b)** *xref*-linked by an ontology

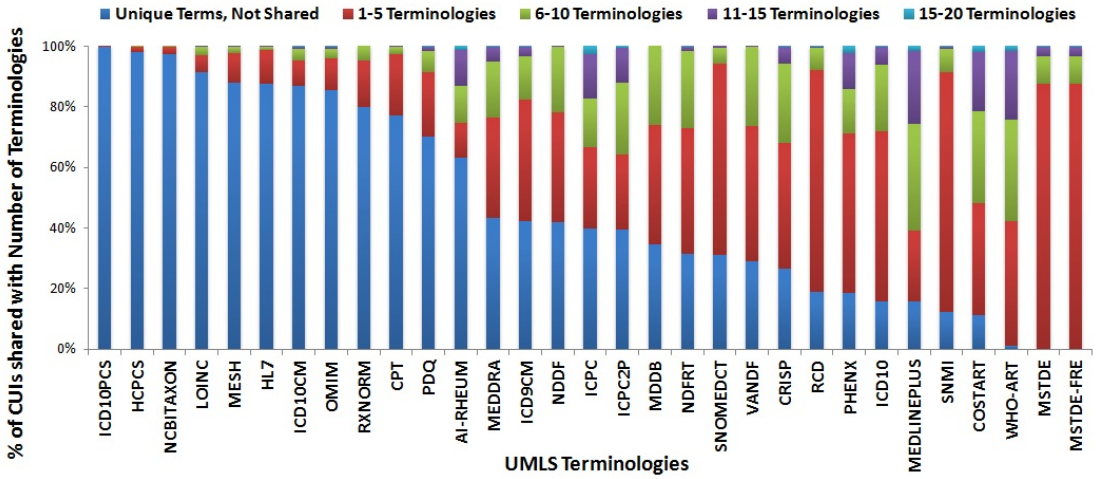


Fig. 2. Percentage (%) of CUIs in each UMLS terminology shared with other terminologies

the resource identifier from each term IRI, we removed those terms which had almost similar term IRIs (having the same identifier and source ontology, but a different or incorrect representation), and the list reduced to 93,650 term labels (1.6%). The last step does not represent actual reuse between ontologies, but rather that ontology developers showed an intention to reuse terms, but used different and sometimes incorrect term representations (discussed below).

Force-directed network visualization We developed an interactive force-directed network visualization, using the third statistic, where the ontologies form the nodes of the networks and the edges connecting them indicate the extent of term reuse between them. The nodes are colored according to the group under which the ontology falls, and the size of the nodes depends on the total number of terms in the current version of the ontology. The thickness of the edge is proportional to the total number of terms shared between the connected ontologies, and the colour varies according to the construct. The graph can be constrained by hovering over any node, to display only the directly related nodes. The interactive version can be accessed at: `http://stanford.edu/~maulikrk/apps/OntologyReuse/`.

The detailed results are available at `http://stanford.edu/~maulikrk/data/OntologyReuse/`.

5 DISCUSSION

Ghazvinian *et al.* (2011) outlined the consistent term overlap, yet minimum term reuse, in OBO Foundry ontologies, and commented on the limitations and challenges to achieve *"orthogonality"*. Five years later, evaluating term reuse over the entire continuum of biomedical ontologies (including UMLS terminologies), we see that we are still very far from achieving desirable term reuse. Most ontologies exhibit considerably less than 5% reuse or no reuse through any constructs, and generally reuse terms from only a small set of ontologies. Table 2 lists many of the OBO Foundry member ontologies. The OBO Foundry mandates reuse by candidate ontologies from the member ontologies under its orthogonality aim. However, there is still substantial *term overlap* present among biomedical ontologies, including OBO Foundry ontologies. *Term overlap*, in itself, is not a good indicator of potential term reuse, as there may be terms in different ontologies which are lexically similar, but represent different concepts (e.g., similar anatomical concepts between Zebrafish Anatomy (ZFA) and Xenopus Anatomy (XAO)), and lexically-different terms may represent the same concept (e.g., *myocardium* and *cardiac muscle*). Hence rigorous methods to detect contextual term overlap are required.

By examining the terms that shared the same labels, we found various IRI patterns that could indicate that the ontology developers showed the intention to reuse terms (same identifiers and source ontologies). These patterns were not considered as term reuse as the IRIs used different representations for the same terms, and no explicit CUI or *xref* mappings were found. Hence, the advantages of term reuse can not be experienced. On using the right IRI representation, the term overlap could reduce substantially. We describe the three most prevalent patterns below.

Different versions: SAO and SOPHARM reuse terms from BFO version 1.0, whereas the majority of other ontologies reuse the corresponding terms found in version 1.1. As mentioned in Section 3, CSEO and SYN reuse terms from an older version of NCI Thesaurus. For example, we found `NCIT:Cerebral_Vein` instead of the recent `NCIT:C53037`.

Different notations: Terms reused from FMA were referenced in multiples ontologies using different notations without consistency or interlinks. For example, `OBO:FMA_31396` is reused as `OBO:owlapi/fma#FMA_31396`, `OBO:owl/FMA#FMA_31396`, and even with the entire label `OBO:fma#Cartilage_of_inferior_surface_of_posterolateral_part`.

Different namespaces: Different ontologies tend to use completely different namespaces for the source ontology. For example, RH-MESH uses `http://phenomebrowser.net/ontologies/mesh/mesh.owl`, while most other ontologies reuse `http://purl.bioontology.org/ontology/MESH`. We also found reuse of SNOMED CT terms with two distinct namespaces: `http://ihtsdo.org/snomedct/clinicalFinding` and `http://purl.bioontology.org/ontology/SNOMEDCT`.

There are direct (semantic interoperability, cost reduction) and indirect (EHR mining, query federation) advantages of term reuse. In the Linked Open Dataspace, newer, collaborative efforts, such as Bio2RDF (Callahan *et al.*, 2013), provide strict guidelines for the representation of concept identifiers while publishing data as RDF. ProtégéLov (Garcia-Santa *et al.*, 2015) allows reuse of terms directly from the Linked Open Vocabularies repository using `owl:equivalentClass` and `rdf:subClassOf` axioms.

Our analysis indicate that while ontology developers may exhibit an intention for term reuse, the lack of guidelines and semi-automated tools for ontology term reuse seem to hinder these goals. Our visualization of reuse dependencies could guide developers to reuse terms in their own ontology based on the structure of ontologies in related domains. Identifying reuse patterns and providing personalized recommendations during the development phase could help increase term reuse and reduce term overlap. Incorporating a reuse module in ontology editing tools could also keep developers updated when the representation of the source term changes.

6 CONCLUSION

We analyzed the extent of term reuse and overlap in 377 biomedical ontologies from BioPortal along three reuse constructs: explicit reuse, *xref* reuse, and CUI reuse. Despite the considerable level of overlap (14.4%), there is very little reuse ($< 5\%$) among biomedical ontologies, both at the level of an ontology and at the level of individual terms. We developed a force-based visualization that helps users to understand the reuse dependencies across different ontologies. We also identified error patterns in applying reuse that we discovered in our empirical analysis. Our future work includes research on identifying reuse patterns in an empirical way, and building a recommendation module for the Protégé toolset that would suggest terms that have been reused together with existing terms. Our strong belief is that better guidelines and tool support will enhance the reuse among biomedical ontologies.

ACKNOWLEDGMENTS

The authors acknowledge Manuel Salvadores Olaizola for providing a triplestore dump of BioPortal ontologies. This work is supported in part by grants GM086587 and GM103316 from the US National Institutes of Health.

REFERENCES

Alexander, C. Y. (2006). Methods in biomedical ontology. *Journal of biomedical informatics*, **39**(3), 252–266.

Bodenreider, O. (2004). The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic acids research*, **32**(suppl 1), D267–D270.

Bodenreider, O. (2008). Biomedical ontologies in action: role in knowledge management, data integration and decision support. *Yearbook of medical informatics*, page 67.

Bontas, E. P. *et al.* (2005). Case studies on ontology reuse. In *Proceedings of the IKNOW05*, volume 74.

Callahan, A. *et al.* (2013). Bio2RDF release 2: Improved coverage, interoperability and provenance of life science linked data. In *The Semantic Web: Semantics and Big Data*, pages 200–212. Springer.

Corcho, O. *et al.* (2003). Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data & knowledge engineering*, **46**(1), 41–64.

Courtot, M. *et al.* (2011). MIREOT: The minimum information to reference an external ontology term. *Applied Ontology*, **6**(1), 23–33.

d'Aquin, M. *et al.* (2009). Criteria and evaluation for ontology modularization techniques. In *Modular ontologies*, pages 67–89. Springer.

Garcia-Santa, N. *et al.* (2015). *Protege LOV Plugin*. http://goo.gl/9fmTf7 (accessed March 05, 2015).

Ghazvinian, A. *et al.* (2011). How orthogonal are the OBO foundry ontologies? *J. Biomedical Semantics*, **2**(S-2), S2.

Hanna, J. *et al.* (2012). Simplifying MIREOT: a MIREOT protégé plugin. In *The Semantic Web– ISWC 2012*.

Kamdar, M. R. *et al.* (2014). ReVeaLD: A user-driven domain-specific interactive search platform for biomedical research. *Journal of biomedical informatics*, **47**, 112–130.

Matentzoglu, N. *et al.* (2013). A snapshot of the OWL web. In *The Semantic Web–ISWC 2013*, pages 331–346. Springer.

Nair, J. (2014). *BioPortal Import Plugin*. http://goo.gl/LL75TR (accessed March 01, 2015).

Noy, N. F. *et al.* (2001). Creating semantic web contents with protege-2000. *IEEE intelligent systems*, **16**(2), 60–71.

OBOFoundry (2011). *Inter-ontology Links*. http://goo.gl/OSrSjP (accessed March 01, 2015).

Pathak, J. *et al.* (2009). Survey of modular ontology techniques and their applications in the biomedical domain. *Integrated computer-aided engineering*, **16**(3), 225–242.

Poveda Villalón, M. *et al.* (2012). The landscape of ontology reuse in linked data. In *Proceedings of OEDW 2012*. Informatica.

Rubin, D. L. *et al.* (2008). Biomedical ontologies: a functional perspective. *Briefings in bioinformatics*, **9**(1), 75–90.

Smith, B. *et al.* (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, **25**(11), 1251–1255.

Tudorache, T. *et al.* (2010). Ontology development for the masses: creating ICD-11 in WebProtégé. In *Knowledge Engineering and Management by the Masses*, pages 74–89. Springer.

W3C (2012). *OWL 2 Web Ontology Language Document Overview*. http://www.w3.org/TR/owl2-overview/ (accessed March 01, 2015).

Wächter, T. *et al.* (2011). DOG4DAG: semi-automated ontology generation in obo-edit and protégé. In *Proceedings of SWAT4LS 2011*, pages 119–120. ACM.

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**(suppl 2), W541–W545.

Xiang, Z. *et al.* (2010). OntoFox: web-based support for ontology reuse. *BMC research notes*, **3**(1), 175.

Aboutness: Towards Foundations for the Information Artifact Ontology

Barry Smith[1,*] and Werner Ceusters[2]
[1] Department of Philosophy, University at Buffalo, 126 Park Hall, Buffalo, USA
[2] Department of Biomedical Informatics, University at Buffalo, 921 Main Street, Buffalo, USA

ABSTRACT
The Information Artifact Ontology (IAO) was created to serve as a domain-neutral resource for the representation of types of information content entities (ICEs) such as documents, data-bases, and digital images. We identify a series of problems with the current version of the IAO and suggest solutions designed to advance our understanding of the relations between ICEs and associated cognitive representations in the minds of human subjects. This requires embedding IAO in a larger framework of ontologies, including most importantly the Mental Functioning Ontology (MFO). It also requires a careful treatment of the aboutness relations between ICEs and associated cognitive representations and their targets in reality.

1 INTRODUCTION

At the heart of the IAO is the term 'Information Content Entity' (ICE), which is currently defined as follows:

> INFORMATION CONTENT ENTITY =def. an ENTITY which is (1) GENERICALLY DEPENDENT on (2) some MATERIAL ENTITY and which (3) stands in a relation of ABOUTNESS to some ENTITY.

An ICE is thus conceived as an entity which is about something in reality and which can migrate or be transmitted (for example through copying) from one entity to another. In what follows we introduce and defend proposals to improve this definition and the IAO as a whole.

The relation of generic dependence was introduced into BFO 1.1 in order to capture the fact that some dependent entities – for example the dependent entity which is the pattern of ink marks in your copy of the novel *War and Peace* (a complex *quality* in BFO terms) – are able to migrate from one bearer to another (e.g. through use of a photocopier). Generic dependence can thus be defined as follows:

> *a* ***generically depends on*** *b* = def. *a* exists and *b* exists and: for some universal B, *b* ***instance_of*** B and necessarily (if *a* exists then some B exists)

In BFO 1.0 the migration of dependent entities from one bearer to another was excluded. Dependence was seen as amounting in every case to *specific* dependence, or in other words as a relation which obtains between one entity and another *specific* entity when the first is of its nature such that it cannot exist unless the second also exists. A smile is dependent in this sense on a certain specific face, a headache on a certain specific head, a charge on a certain specific conductor. Generic dependence, in contrast, obtains where the first entity is dependent, not on some *specific* second entity, but rather merely on there being *some* second entity of the appropriate type (Smith et al. 2015). A DNA sequence is generically dependent in this sense on *some* but not on any *specific* DNA molecule; a pdf file on *some* but not on any specific memory store; and so on.

A generically dependent entity is in each case *concretized* (see definition in section 5) in some specifically dependent entity (more specifically in some BFO:quality). For example, this DNA sequence is concretized in this specific ordering (pattern) of nucleotides in this particular molecule; this sentence is concretized in this pattern of ink marks on this piece of paper (or also in this pattern of neuronal connections in the brain of the subject who reads it). The term 'pattern' can thus be understood in two senses – as referring either (i) to what is shared or communicated (between original and copy, between sender and receiver), or (ii) to the specific pattern before you when you are reading from your copy of Tolstoy's novel.

We can now define:

> INFORMATION QUALITY ENTITY (IQE) =def. a QUALITY that is the concretization of some INFORMATION CONTENT ENTITY (ICE) (Smith et al., 2013),

noting that IQEs are called 'information carriers' in the current version of IAO.

All concretizations are qualities in the BFO framework. Such qualities can serve as the basis for dispositions. When we concretize a lab test order by reading the text of the order on our screen, then in addition to the mental quality that is formed in our mind as we read the text, there is also a disposition to be realized in our actions of carrying out the relevant test. This disposition may come into being simultaneously with the mental quality created through our understanding of the text, but it is still dependent on this quality, as is shown by the fact that the latter may exist even in the absence of any accompanying disposition.

We define 'artifact' and 'information artifact' as follows:

> ARTIFACT =def. a MATERIAL ENTITY created or modified or selected by some agent to realize a certain FUNCTION or ROLE (Examples: a key, a lock, a screwdriver)

> INFORMATION ARTIFACT =def. an ARTIFACT whose function is to bear an INFORMATION QUALITY ENTITY. (Ex-

* To whom correspondence should be addressed: phismith@buffalo.edu

amples: a hard drive, a traffic sign, a printed form, a passport, a currency note, an RFID chip, a SIM card)

As a matter of definition, therefore, all information artifacts are material entities. While every ICE is dependent upon some material entity that is its bearer ICEs themselves are not material entities.

In reflection of the needs it was originally designed to address, the IAO is focused deliberately on ICEs associated with information *artifacts* – above all scientific publications and databases – thus with information entities which are continuants in BFO terms. No less important, however, is the occurrent side of the informational coin, which is made up of those processes – above all acts of thinking, speaking, hearing, writing and reading – through which ICEs are created, understood, and communicated. Given that thinking and speaking pre-dated writing, we know that acts of these sorts existed long before there were any information artifacts. They are of crucial importance to the ontological treatment of the phenomenon of aboutness because it is they which provide the relational tie between representations and their targets in reality.

If, therefore, we are to deal with these more fundamental aspects of the information pipeline, then we will need to embed the IAO into a wider framework of ontologies. This would include, on the one hand, all existing domain ontologies, which can be seen as representing the portions of reality about which we have information – they are ontologies of the various families of *targets* of aboutness. More importantly here, however, it would include on the other hand the Mental Functioning Ontology (MFO), which is designed to provide the resources to describe different types of cognitive acts, including those cognitive acts as a result of which ICEs are created (Ceusters & Smith, 2010).

2 ABOUTNESS AND PORTIONS OF REALITY

Aboutness corresponds to what is otherwise referred to by means of the expressions 'reference' or 'denotation,' (Yablo, 2014) but generalized to include not merely linguistic reference but also the relations of cognitive or intentional directedness that are involved, for instance, when a nurse is *measuring a patient's pulse rate* or a doctor is *observing a rash on a patient's thigh*. These processes are about, respectively, a *pulse* and a *rash*. When the nurse enters the string *72 beats per minute* in the medical chart of the patient, then there is an ICE that is concretized in the ink (or pixel) pattern exhibited on the chart, which inherits its aboutness from the aboutness of what we shall call the nurse's *direct cognitive representation* of the pulse. The latter is a (binary) relational quality; it links the nurse causally to the target of his observations. It is on this basis that, by entering data, he creates an ICE that is also tied relationally to its target in reality. Thus the ICE is not an abstract entity analogous to a 'proposition' in logical parlance. Rather it is a created, historical entity that is marked by the feature of indexicality: its aboutness and its rootedness in time and context are analogous to those of an instruction issued by someone who points his index finger and says 'go *there now*.'

The current IAO definition of ICE can account for the aboutness involved in many examples of these sorts. However, we believe that it falls short when it comes to more complex cases. In (Ceusters, 2012) we proposed broadening the definition of ICE to require 'aboutness to some *portion of reality*' rather than just 'to some *entity*,' in order to allow the domain of the aboutness relation to include *inter alia*

- universals, for instance in the ICE concretized by the string *there are no instances of dinosaur which survive*,
- relations, for instance in the ICE concretized by the string *the part-whole relation is transitive*,
- other ICEs, for instance when someone asserts that what someone else just stated is true, and
- configurations, for instance in the ICE concretized by *Barack Obama is the current President of the USA*

– none of which is an entity in BFO terms.

The last example on this list is not only about Barack Obama but also about his *role* of being President of the USA and about the USA itself. But it is not only about these entities taken singly; in addition, it is about how the three entities are related to each other in a certain interval of time, and about the entire portion of reality – the configuration – made up by all of these together. This configuration is *asserted to exist* by a human subject using the corresponding sentence in a specific sort of context and with a specific sort of associated cognitive quality. But it can also be *referred to*, for instance when someone makes a second-order assertion using a nominalized expression, as in: *That Barack Obama is President of the USA is of epoch-making significance*.

3 INFORMATION AND MIS-INFORMATION

We can on this basis address another issue with IAO's current definition of ICE, which is that it does not give us a clear way of doing justice to the distinction between *information* on the one hand and what we might call *mis-information* on the other. Consider the ICE concretized in the sentence *Barack Obama was never President of the USA*, written on some piece of paper in 2015. This ICE is indeed about Barack Obama, the USA, and so forth. But what it communicates about these entities is something that is false. Our amended definition of ICE can allow us to accept that both information and mis-information exist, but also to recognize that the latter is not a special type of the former (that what some people might call 'false information' is not a special type of information, any more than a cancelled oophorectomy is a special type of oophorectomy). We achieve this by using our generalized definition of ICE to formulate a view according to which the relation of aboutness between a composite (for example sentential) ICE

and the associated portions of reality can obtain (or fail to obtain) simultaneously on two (or in principle more than two) levels: first, on the level of simple referring expressions such as 'Barack Obama' and 'USA'; and second, on the level of more complex expressions such as sentences and their nominalizations.

A true sentence on the upper level is about a corresponding configuration (where the term 'configuration' is to be understood in a way similar to the way 'fact' or 'obtaining state of affairs' are understood by some philosophers (Wittgenstein, 1961)). We can now capture the fact that a given compound expression may inherit aboutness from some or all of its constituent simpler referring expressions but fail in its claim to aboutness (and thus to convey information) when taken as a whole.

If someone writes on a piece of paper the sentence *Barack Obama is President of Russia*, then there is an ICE – concretized by this written string and by any copies made thereof – which is generically dependent on the piece of paper and which is about (on the aforementioned lower level) Barack Obama, his being president, and Russia. But this ICE is not about any corresponding configuration, simply because there is no corresponding configuration. It is for this reason that the given sentence, while it is *about* certain entities in reality, is nonetheless not *true of* those entities. This strategy can be used also to explain how a fictional sentence such as *Sherlock Holmes was a user of cocaine*, can concretize an ICE – by inheriting aboutness from one or more of its components (here for example the string *cocaine*, which is about a corresponding universal) – even though the sentence as a whole is not about anything in reality.

A related problem with the current IAO is that it does not provide us with the resources to do justice to what happens with certain types of ICE when what they are about changes over time. The problem here is that the ICE concretized by the sentence *Barack Obama was never President of the USA* written on a piece of paper in 2007 was true when it was written; yet it appears that this very same sentence, when read by some observer in 2009, would be false.

This appearance is misleading, however, for it is not the case that the ICE in question changes in the intervening period. Rather, what has changed is the first-order reality that this ICE claims to be about. Certainly as a result of these changes in first-order reality there came into existence many new ICEs relevant to Obama, the presidency and the USA, with many new concretizations. But the original ICE, with its original concretization born with its original act of creation, must nonetheless still be evaluated as true. This is because, as in the case of the nurse's data entry above, the ICE in question has its time of origin baked into it through the indexicality of the *was* in *was never President*.

We shall presuppose in what follows that information artifacts do not bear information in and of themselves, but only because cognitive subjects associate representations of certain sorts with the patterns which they manifest. We thus view the aboutness that is manifested by information content entities in accordance with the doctrine of the 'primacy of the intentional' (Chisholm, 1984), according to which the aboutness of those of our representations formulated in speech or writing (or in their printed or digital counterparts) is to be understood by reference to the cognitive acts with which they are or can in principle be associated. The entry *72 beats per minute* is about what it is about because of what the nurse himself directly observed when he measured the patient's pulse (or, in the case where the ICE is created by sensor devices automatically adding data to the chart, it is about what the nurse would have observed in the given circumstances).

At higher levels we may have ungrounded representations, as illustrated for example in the letter published by Urbain Le Verrier in 1859 (Le Verrier, 1859) in which there appears an intended reference to a planet that is asserted to be intermediate between Mercury and the Sun, a planet which in 1860 Le Verrier baptised 'Vulcan'. This intended reference depended on a certain belief on Le Verrier's part in the existence of an intra-Mercurial planet. When we understand Le Verrier's text today, however, then we have a different sort of cognitive representation – involving what we refer to below as a *recognized non-referring representational unit* (RNRU) – in which this intended reference to a planet has been cancelled.

Such changes in our understanding of the reference of terms are of course a common phenomenon in the world of ontology, and specifically in the world of ontology versioning. Paying careful attention to these changes forms the basis for the strategy for ontology evaluation we have outlined in (Ceusters & Smith, 2006).

4 REPRESENTATION AND REFERENCE

We build on the notions of representation and representational unit informally introduced in (Smith et al., 2006). A representation is there described as *an idea, image, record, or description which refers to (is of or about), or is intended to refer to, some entity or entities external to the representation*. Note that 'representation' is thus more comprehensive in scope than 'ICE,' even on our proposed more inclusive definition of the latter, since an ICE must in every case be *about* some portion of reality, where the aboutness in question must always be veridical, so that 'being about' is a success verb. A representation, in contrast, is required merely to intend to be about something, and this intention might fail (as when a child draws what she thinks of as a unicorn).

We provided a formal definition of 'representation' along these lines in (Ceusters & Smith, 2010):

> **REPRESENTATION** =def. a QUALITY which ***is_about*** or is intended to ***be about*** a PORTION OF REALITY (POR).

We can now single out cognitive representations (representations of the sorts instantiated in the brains of beings like ourselves) by means of the terms:

> MENTAL QUALITY =def. a QUALITY which ***specifically depends on*** an ANATOMICAL STRUCTURE in the cognitive system of an ORGANISM.
>
> COGNITIVE REPRESENTATION =def. a REPRESENTATION which is a MENTAL QUALITY.

defined in the Mental Functioning Ontology. We are here attempting to remain neutral as concerns the precise nature of cognitive representations; thus it does not follow from the definitions that such representations involve something like images; nor does it follow that they must all be *conscious* representations.

As concerns occurrents in the realm of cognition, it is clear that mental processes, too, for example processes of thinking or imagining or remembering, may be about or be intended to be about some portion of reality. We hypothesize, however, that such occurrent representations are always such as to inherit their intended aboutness from some underlying continuant representation. When the doctor sees, and recognizes, for example, that there is a rash on her patient's leg, then her act of recognition coincides temporally with the beginning to exist of a correspondingly targeted (relational) mental quality on her part (Smith, 1987).

As we saw above, cognitive representations may be more or less complex. When analyzed into their constituent parts, however, then we arrive at what we called 'representational units' (RUs), defined as *the smallest constituent sub-representations*, including *icons, names, simple word forms, or the sorts of alphanumeric identifiers we might find in patient records*. (Smith et al., 2006)

Subtypes of representational unit can then be defined as follows (Ceusters & Smith, 2010):

1. *Referring representational unit* (RRU)*:* an RU which is both intended to be about something and does indeed succeed in this intent.
2. *Non-referring representational unit* (NRU)*:* an RU which, for whatever reason, fails to be about anything.
3. *Unrecognized non-referring representational unit* (UNRU): an NRU which, although non-referring, is intended and believed to be about something;
4. *Recognized non-referring representational unit* (RNRU): an NRU which was once intended and believed to be about something, but which, as a result of advances in knowledge, is no longer believed to be so;
5. *Representational unit component* (RUC): a component of a representation that is not intended by the artifact's authors to refer in isolation;

RU	'Paris'
NRU	'Atlantis'
UNRU	'Vulcan' (as used by Le Verrier in 1860)
RNRU	'Vulcan' (as used now when referreing to Le Verrier's error)
RUC	'Le' (as it appears in the third row of this table)

Table 1: Examples of types of representational unit

Note that, as the 'Vulcan' case makes clear, classifications of representations under headings 1. to 5. may change with time. Note, too that, while items 2. to 5. on this list signify one or other kind of shortfall from aboutness, representations under item 1. include the fundamental (grounding, target-securing) cases of *direct cognitive representation* referred to in the case of the nurse taking someone's pulse as in our example above.

5 PROPOSAL

5.1 Primitives and elucidations

To do justice formally to the foregoing we propose the following primitive relational expressions. These cannot be defined, but only elucidated by means of examples and informal specifications of their meanings.

x ***is_about*** *y* means**:**

> *x refers to or is cognitively directed towards y.* **Domain:** representations; **Range:** portions of reality. **Axiom:** if *x* ***is_about*** *y* then *y* exists (veridicality).

x ***concretizes*** *y* ***at*** *t* means:

> *x* is a QUALITY & *y* is a GENERICALLY DEPENDENT CONTINUANT
>
> & for some material entity *z*, *x* ***specifically_depends_on*** *z* at *t* & *y* ***generically_depends_on*** *z* at *t*
>
> & if *y* migrates from bearer *z* to another bearer *w* then a copy of *x* will be created in *w*.

x ***is_a_direct_cognitive_representation_of*** *y* means:

> *x* is a COGNITIVE REPRESENTATION in some subject *s*
>
> & *x* ***is_about*** *y* & *x* comes into existence, as a result of a causal process initiated by *y* and in a way appropriate to *y*, in the cognitive system of *s*. **Example:** a causal process of visual perception initiated by an object presented visually to *s*.

5.2 Definitions

x ***is_a_representation_of*** *y* =def. *x* is a REPRESENTATION & *x* ***is_about*** *y* (where *y* is a portion of reality). Note that not all representations are about something.

x ***is_conformant_to*** *y* =def. *x* is an INFORMATION QUALITY ENTITY & *y* is a COGNITIVE REPRESENTATION & there is some GDC *g* such that *x* ***concretizes*** *g* and *y* ***concretizes*** *g*. **Example:** *x* is a sentence on a piece of paper, *y* is the

belief of the author of the sentence who wrote the sentence as an expression of her belief, and *g* is the ICE (the content) that belief and sentence share.

6 DISCUSSION

Although it is a requirement that the target of aboutness be a portion of reality (POR), there is no requirement that the relevant POR exists at the time when the associated cognitive representation exists. Thus a patient can contemplate a past disorder, for instance by regretting his not having accepted the advice of some clinician. His thoughts are then about that very disorder, and not for example about his memories thereof. This is so independently of whether the nature of the disorder is known to him or not.

There is also no requirement that the agent of a veridical representation knows what the portion of reality is that his representation is about: even a baby, or a cat, may see a flow cytometer. We can directly represent an object even though we are ignorant of or mistaken about what universal it instantiates.

There is also – as is illustrated by the case of believers in the Higgs boson before there was evidence for its existence – no requirement that aboutness must imply that the subject *knows* that what he is representing exists – he must merely *believe* that it exists.

Although neuroscience, to our best understanding, is not yet sufficiently advanced to provide answers to the question what the precise physical basis of a mental quality exactly is – for example whether it is certain spatial configurations of one or more molecules in one or more brain cells – we believe that the following hypothesis is correct: that an anatomical structure in which there *can* inhere a mental quality need not always *have* a mental quality inhering in it (in this respect *having a mental quality* is comparable to having the quality of *being pregnant* and is to be contrasted with qualities such as height and mass, given that something in which there can inhere a height or a mass must always have a height or mass of some determinate sort). From this, it is then just a short step to the question of whether there can be unconscious representations, a question which, however, we must here leave aside for reasons of space.

7 CONCLUSION

IAO was designed to deal with information artifacts, which is to say with continuants such as the information stored in hard drives or formulated in written sentences or in printed texts – thus with information that is shareable between multiple bearers, including bearers existing at different times. As will by now be clear, the IAO must be embedded in a broader framework of ontologies, including the Mental Functioning Ontology (Hastings *et al., 2012)*. In the future we must address for example how an agent can use sight (or, in the case of Braille, touch) to process concretization in such a way as to generate mental representations that are conformant to the associated ICEs. For this we will require a Language Ontology – extending the Ontology of Document Acts proposed in (Almeida, et al. 2012) – that will allow us to do justice to the ways in which sentences can be not merely believed and thought but also asserted, heard, seen (for example in the case of sign language), understood, and formulated in written or printed texts.

ACKNOWLEDGEMENTS

We are grateful to Bill Duncan, Mark Jensen, Tatiana Malyuta, Ron Rud-nicki, Alan Ruttenberg and Selja Seppälä for many valuable discussions.

REFERENCES

Almeida, M.B., Slaughter, L., & Brochhausen, M. (2012). Towards an ontology of document acts: Introducing a document act template for healthcare. Lecture Notes in Computer Science, 7567, 420-425.

Ceusters, W. (2012). An information artifact ontology perspective on data collections and associated representational artifacts. Stud Health Technol Inform, 180, 68-72.

Ceusters, W., & Smith, B. (2006). A realism-based approach to the evolution of biomedical ontologies. AMIA Annu Symp Proc, 121-125.

Ceusters, W., & Smith, B. (2010). Foundations for a realist ontology of mental disease. Journal of Biomedical Semantics, 1(10), 1-23. doi: doi:10.1186/2041-1480-1-10

Chisholm, R. M. (1984). The primacy of the intentional. Synthese, 61(1), 89-109. doi: Doi 10.1007/Bf00485490

Hastings, J., Ceusters, W., Jensen, M., Mulligan, K., & Smith, B. (2012). Representing mental functioning: Ontologies for mental health and disease Towards an ontology of mental functioning (icbo workshop), proceeedings of the third international conference on biomedical ontology.

Le Verrier, U. (1859). Lettre de m. Le verrier à m. Faye sur la théorie de mercure et sur le mouvement du périhélie de cette planète. Comptes rendus hebdomadaires des séances de l'Académie des sciences (Paris), 49, 379-383.

Smith, B. (1987). On the cognition of states of affairs. In K. Mulligan (Ed.), Speech act and sachverhalt (Vol. 1, pp. 189-225): Springer Netherlands.

Smith, B et al. (2015). Basic formal ontology 2.0 draft specification and user manual. from http://bfo.googlecode.com/svn/trunk/docs/bfo2-reference/BFO2-Reference.docx

Smith, B., Kusnierczyk, W., Schober, D., & Ceusters, W. (2006). Towards a reference terminology for ontology research and development in the biomedical domain Kr-med 2006, biomedical ontology in action. Baltimore MD, USA

Smith, B., Malyuta, T., Rudnicki, R., *et al*. (2013). Iao-intel: An ontology of information artifacts in the intelligence domain. Paper presented at the STIDS.

Wittgenstein, L. (1961). Tractatus logico-philosophicus. London: Routledge and Kegan Paul.

Yablo, S. (2014). Aboutness, Princeton, NJ:: Princeton University Press.

Medical and Transmission Vector Vocabulary Alignment with Schema.org

William Smith*, Alan Chappell, and Courtney Corley
Pacific Northwest National Laboratory

ABSTRACT

Available biomedical ontologies and knowledge bases currently lack formal and standards-based interconnections between disease, disease vector, and drug treatment vocabularies. The PNNL Medical Linked Dataset (PNNL-MLD) addresses this gap. This paper describes the PNNL-MLD, which provides a unified vocabulary and dataset of drug, disease, side effect, and vector transmission background information. Currently, the PNNL-MLD combines and curates data from the following research projects: DrugBank, DailyMed, Diseasome, DisGeNet, Wikipedia Infobox, Sider, and PharmGKB. The main outcomes of this effort are a dataset aligned to Schema.org, including a parsing framework, and extensible hooks ready for integration with selected medical ontologies. The PNNL-MLD enables researchers more quickly and easily to query distinct datasets. Future extensions to the PNNL-MLD may include Traditional Chinese Medicine, broader interlinks across genetic structures, a larger thesaurus of synonyms and hypernyms, explicit coding of diseases and drugs across research systems, and incorporating vector-borne transmission vocabularies.

1 INTRODUCTION

Medical vocabularies and ontologies have been developed over the last two decades and represent a large cross-section of Linked Open Datasets. Several research initiatives are now de facto authoritative data stores used by thousands of medical researchers daily including: DrugBank (Law, et al. 2014), PharmGKB (Stanford University 2014), Vectorbase (National Institute of Allergy and Infectious Diseases; National Institutes of Health; Department of Health and Human Services 2014), Uniprot (Consortium 2014), Allen Institute for Brain Science (AIBS) Brain Map (Allen Institute for Brain Science 2014), and Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa, et al. 2014). However, with the collection of these advanced medical vocabularies and descriptive logic rules, a data classification divergence occurred.

Medical research groups rarely attempted to standardize vocabularies and ontologies with other research teams. This created data resources that are not natively interconnected with knowledge bases outside of a specific research objective. Furthermore, specific medical coding may exist on an entity level (OMIM, MeSH, eMedicine, etc), but there is no inherent guarantee across data sources that these codes are available or properly represented in a standard format. Entity matching between datasets is complicated by the fact most medical classes operate on a complex set of synonyms, hypernyms or taxonomical naming schemas that typically are not standardized across research projects and communities.

This effort addresses the tracking of a disease and treatment regimen across vector-borne transmission variables, including geography and species. The variety of issues described renders any available single source of research data unusable to address realistic research questions across the breadth of this domain space. Table 1 represents common diseases and transmission vectors for tracking vector-borne infections that were used as the starting point.

Disease	Transmission Vector
Eastern Equine Encephalitis Virus	Culiseta melanura / Cs. morsitans
Western Equine Encephalitis Virus	Culex / Culiseta
Highlands J Virus	Culiseta melanura
St. Louis Encephalitis Virus	Culex
West Nile Virus	Many
La Crosse Encephalitis	Ochlerotatus triseriatus synonym Aedes triseriatus
Chikungunya	A. albopictus and A. aegypti
Dengue Fever	Genus Aedes, principally A. aegypti

Table 1. Common diseases and associated transition vectors.

2 INITIAL VOCABULARIES AND ONTOLOGIES

One way of making use of the extensive previous work in disease descriptions by different research efforts and enabling associations across these vocabularies is assembling a knowledge base targeting the research area of interest. The more overlapping sets of information present in the resulting knowledge base the better chance a system has of making associations across vocabularies simply because of the availability of information on which to make the associations. For tracking vector-borne infections, disease datasets

* To whom correspondence should be addressed: william.smith@pnnl.gov

Dataset	Schema Entities	Schema Predicates	Schema Objects	Unaligned Entities	Unaligned Predicates	Unaligned Objects
Diseasome	4,213	7	31,538	3,938	13	43,836
PharmGKB	3,442	2	43,030	0	3	10,326
DisGeNet	13,172	1	13,172	0	3	39,516
Wikipedia Infobox	2,273	2	5,747	0	3	5,179
DailyMed	5,019	3	11,729	9,294	25	151,243
DrugBank	4,772	10	155,410	19,686	89	29,230
Sider	2,661	10	51,244	9	89	32,370

Table 2: Entity, predicate, and object counts after Schema.org alignment.

are a primary focus. Therefore, the team initially collected authoritative resources with a large amount of disease entities and extensive properties attached to each entity. The chosen datasets and entity count estimates include: Diseasome (Goh, et al. 2007), PharmGKB, DisGeNet (DisGeNet 2014), and Wikipedia Infobox (Wikimedia Foundation 2014). Table 2 depicts the data sets incorporated and the scale of the associated relevant vocabularies. These datasets provided different levels of expression across diseases, an example being PharmGKB having a small number of diseases with many properties versus DisGeNet having several times more entities expressed with a single name property and medical code.

Drug datasets, while initially not appearing to be part of the use case of tracking vector-borne infections, are useful as a direct path for aligning diseases across naming conventions. The selected drug datasets and estimated entity counts include: DailyMed (United States National Library of Medicine 2014) and DrugBank. In practice, drug datasets contain an extensive listing of medical codes, collected from prior research, across databases often missing from disease datasets. While these codes can be imprecise, they provide a starting point for entity interlinks and additional data enrichment through NLP and Linked Data techniques. When we focus on the disease medical codes affected by a specific treatment, the medical codes in the drug datasets enable us to programmatically create *owl:sameAs* relations across diseases in the disease data sets that are missing explicit matching medical codes or proper names. As a result, when drugs listing extensive medical codes are used as a reference point, diseases often can be more fully described, as missing medical codes are combined across datasets for more complete Linked Open Data.

Side effects were also included in the initial PNNL-MLD. This additional information enables detecting symptoms and matching the symptom to a disease or drug combination. The Sider (Kuhn, et al. 2010) dataset was selected as the lone source due to limited availability, but Sider contained dozens of different connections per entity across drugs further helping to align the combined dataset.

3 TARGET VOCABULARY: SCHEMA.ORG

In order to facilitate easier query description through a consistent vocabulary, the project chose one primary vocabulary to encompass the collected data. Selection of this vocabulary is driven by two primary considerations: 1) adequate expressiveness for the queries, and 2) not overly prescriptive such that it creates conflicts with the individual dataset semantics. The selection of this primary vocabulary is important, as it is an opportunity to promote wider use of the assembled dataset through adoption of an impactful or widely used vocabulary.

Schema.org (Google Inc; Microsoft Inc; Yahoo Inc 2014) was released in June 2011, and has become the search industry preferred standard for publishing search engine readable data. After the release of schema.org a RDFS (W3C RDF Working Group 2004) mapping was created and hosted on http://schema.rdfs.org, and this mapping is now a standard for Linked Data research utilizing Schema.org. Finally, at the end of June 2011, Schema.org released an official OWL (W3C OWL Working Group 2012) version of the Schema.org ontology bridging the gap between vocabulary and description logic.

Schema.org provides a base ontology class for medical entities available as a subclass of *Thing* entitled *MedicalEntity*. The subclasses of the *MedicalEntity* class were selected to represent the disease, drug, and side effect entities available within the PNNL-MLD. Table 3 lists the selected sub-classes.

Schema.org Class	Entity
MedicalCondition	Disease
MedicalCause	Disease Cause
MedicalSignOrSymptom	Disease Symptom
MedicalTherapy, Drug	Drug
MedicalCode	Entity Code
MedicalEntity	Side Effect

Table 3. Schema.org classes selected to represent use case entities.

4 VOCABULARY ALIGNMENT

Simply adding a primary vocabulary to the datasets is not adequate to simplify querying. The source datasets must be aligned with the primary vocabulary so that queries will return results that span and integrate all the available information. The central goal in this alignment is to provide a mapping of the source vocabularies to the new primary vocabulary that preserves the semantics of the source but bridges the divergence between the different knowledge representations.

4.1 Base dataset alignment

The project selected the URI: http://beowulf.pnnl.gov/2014/ to serve as the RDF (W3C RDF Working Group 2004) prefix base for all aligned data. We used this new base URI to simplify software development later in the alignment process. Furthermore, all properties were immediately aligned by import dataset, prefix associations demonstrated by the following:

beo-<dataset-name>:propertyName.

By first associating property and class values with an original prefix denoting dataset we could now track properties that were not explicitly aligned to Schema.org. The *rdfs:label* and *owl:sameAs* properties were left unmodified throughout the entire process, and **schema:alternateName** is used to track synonyms of *rdfs:label*.

4.2 Disease dataset alignment

Four large datasets of varying entity counts and properties were the first targets after the base import of the PNNL-MLD. The first substitution took place by converting all unique entity IRIs to a common format:

beo-disease:<disease-id>

We then added the Schema.org declaration of class:

a schema:MedicalCondition

Primary preventions were added to diseases as drug IRIs were detected:

schema:primaryPrevention beo-drug:<original-drug-id>, …

Finally, we use Table 4 to ensure we can match back to online medical resources and unify datasets:

Schema.org Class	Entity
MedicalCode	IRI
MedicalPage	URI
code	Unknown Code Type

Table 4. Alignment of Schema.org classes to medical resources.

4.3 Drug dataset alignment

Two datasets comprised drug metadata and provided interlinks to side effect metadata. These entities were referenced in both disease and side effect datasets as potential treatments (disease) and causes of (side effect). The first substitution took place by converting all unique entity IRIs to a common format:

beo-drug:<disease-id>

We then added the Schema.org declaration of class:

a schema:Drug

Drug, a subclass of *MedicalTherapy*, was selected due to the semantics of the original data. Drugs have the same medical coding standards as Table 4, but the attributes linking the drugs are more abstract including two descriptions of the drug:

schema:potentialAction
schma:description

To link the drug entity to a disease we replace:

beo-drugbank:possibleDiseaseTarget

with:

schema:possibleTreatment

Finally, drugs can interact with each other creating adverse reactions. The DrugBank dataset provides the interconnections for this possibility. We aligned these reactions by creating the entity type:

a beo-drugbank:drug_interactions

And ensuring the new entity has at least two of the following relations:

schema:interactingDrug beo-drug:id

4.4 Side Effect dataset alignment

The single Sider dataset provides the final links to drug entities with each side effect's unique IRI converted to:

beo-interaction:<effect-id>

Then adding the Schema.org declaration of class:

a schema:medicalEntity

Completing the ontology requires one last step linking drugs to side effects with the drug entity property:

schema:seriousAdverseOutcome beo-interaction:<id>

5 QUERY A DISEASE

Using Dengue Fever as an example disease we can now use **schema:MedicalCondition** to query across all of the disease datasets. The SPARQL (W3C SPARQL Working Group 2013) query below locates the available information in the combined dataset about any medical condition with "dengue" in its name and collects the comments that describe the source of that information.

```
@prefix schema: <http://schema.org/>
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
SELECT ?label ?comment
WHERE {
    ?item a schema:MedicalCondition .
    ?item rdfs:label ?label .
    FILTER (regex(?label, 'dengue', 'i')) .
OPTIONAL { ?item rdfs:comment ?comment }}
```

Running this query on the PNNL-MLD returns Table 5.

?label	?comment
"Dengue shock syndrome"	"Imported from DISGENET"
"Dengue"	"Imported from PharmGKB "
"Dengue Hemorrhagic Fever"	"Imported from PharmGKB"
"Dengue"	"Imported from DISGENET"
"Dengue Hemorrhagic Fever"	"Imported from DISGENET "
"Dengue fever, protection against"	<diseasome>
"Dengue_fever,_protection_against"	<diseasome>

Table 5. Result of SPARQL query on Dengue Fever.

The results in Table 5 expose a current limitation of the system due to regex matching of the label property. Because the query can now reach across several different datasets with conflicting naming schemes an additional normalization process is needed during the data import to normalize labels for all of the entities linked with *owl:sameAs.*

The results in Table 5 show that one simple query now identifies data from three different sources. This begins to show the value of the combined dataset. However, to explore the full impact of the alignment a more complex query is needed that requires the integration of information from multiple sources. Expanding on our previous query we can search across all originally returned "dengue" conditions and append the drug links and treatments added with Schema.org .

```
@prefix schema: <http://schema.org/>
@prefix rdfs: <http://www.w3.org/2000/01/rdf-
schema#>

SELECT DISTINCT ?drugLabel ?diseaseTarget
WHERE {
GRAPH ?G{
    ?item a schema:MedicalCondition .
    ?item rdfs:label ?label .
    FILTER (regex(?label, 'dengue', 'i')) .}
GRAPH ?G1{
    ?item schema:primaryPrevention ?drug .
    ?drug rdfs:label ?drugLabel .}
GRAPH ?G2{
    ?drug schema:possibleTreatment ?target .
    ?target rdfs:label ?diseaseTarget }}
```

This query returns Table 6.

?drugLabel	?diseaseTarget
"Alpha-D-Mannose"	"Dengue_fever,_protection_against"
"Fucose"	"Dengue_fever,_protection_against"

Table 6. Result of SPARQL query for drugs treating Dengue Fever showing value of aligned vocabulary.

Another limitation of the current PNNL-MLD is exposed reviewing the results of Table 6. When creating interlinks across diseases, only the Diseasome entities were referenced in the corresponding drug datasets as possible targets for treatment. To correct this oversight we also need to include *owl:sameAs* associations within our queries, or select a logical reasoner capable of associating and returning all related entities upon a single link between a disease and drug.

Most importantly, Table 6 depicts the value of the combined and aligned PNNL-MLD dataset. Queries like the one given here that require information linking diseases to treatments or symptoms or side effects are now greatly simplified and can focus on a single vocabulary. Schema.org provided classes and properties appropriate for drafting queries that can provide views of the data not visible using only a single source of data.

No technical limitation exists that would restrict a user from loading all of the datasets into separate graphs of an available triplestore and querying the different vocabularies across graphs. However, when we align these datasets into the PNNL-MLD we achieve four major benefits:

1. Queries are now simplified. Early drafts for querying across all of the graphs required queries that were dozens of lines in length, and portions of the queries varied drastically in format and language.
2. A standardized vocabulary, that is industry recognized, is now in place for application development.
3. All of the graphs, when aligned into the PNNL-MLD, are now equally extensible. Adding new vocabularies and ontologies to the original data would require special updates to each dataset, and require updates to each specific portion of a query using that dataset.
4. As shown in Table 2, when a dataset is converted using RDF, and not generated from a different file type (unaligned entities = 0), the Schema.org entities now have a much higher ratio of Schema.org predicates to object triple mappings. By flattening the ontology a simplified query now has access to a much greater range of values and entities.

Additionally, because all modification and additions made while aligning are programmatically defined rather than human expert mediated, new version of the PNNL-MLD can be easily created as source datasets produce new versions.

6 CURRENT LIMITATIONS

The complete PNNL-MLD is now capable of being queried through SPARQL using only Schema.org associations. However, there are still shortcomings in searching for drugs and diseases by name, including the corresponding regex filters. To resolve this conflict a primary label for a group of entities related by *owl:sameAs* should be selected upon entity interlinking with the previous labels turned into **schema:alternateName** properties. Queries should then be composed to either search for a primary name and/or alternate synonym. To remove duplicates imported from different datasets a reasoner capable of merging *owl:sameAs* relations should be used when querying the complete PNNL-MLD.

Medical coding was not at first considered a feature of the application and early versions of the PNNL-MLD did not prioritize accurately creating the properties in Table 3. As it became more apparent diseases and drugs were not consistently labeled across datasets, and outside database entities generally were consistent across datasets, more focus was added to ensure medical codes were applied to drug and disease entities. However, this process was never finalized through Linked Data authentication to ensure the medical codes supplied were accurate for the attached entity.

6.1 Future work

To address current limitations we need to focus on best practices utilizing linked data (Heath and Bizer 2011), and expanding vector transmission geo-properties.

1. Authenticate medical coding. Confirm the entity is correctly aligned to outside sources.
2. Add Gazetteer to provide formal geographic naming entities while also mapping a list of local colloquialisms for geographic regions.
3. Add Vectorbase. (National Institute of Allergy and Infectious Diseases; National Institutes of Health; Department of Health and Human Services 2014)

7 CONCLUSIONS

The broader implications of aligning datasets under a common vocabulary, and making them available using Linked Open Data best practices, is to standardize and expand the original research objectives. When we augment the unique vocabulary and ontology mappings of individual research programs with the broader Schema.org vocabulary, we create data interlinks that enable conceptualization of new questions that bridge the earlier work without requiring replicating research with a broader focus. This combination of separate datasets with common data points aligned to nonexclusive properties and ontology rules simplifies queries, and creates a new superset built for application development and public discovery.

ACKNOWLEDGEMENTS

This work was funded by a contract with the Defense Threat Reduction Agency (DTRA), Joint Science and Technology Office for Chemical and Biological Defense under project number CB10082. Pacific Northwest National Laboratory is operated for the U.S. Department of Energy by Battelle under Contract DE-AC05-76RL01830.

REFERENCES

Allen Institute for Brain Science. *Allen Human Brain Atlas.* 2014. http://human.brain-map.org/ (accessed 2014).

Ashburner, Michael. *BioPortal*. 2014. http://bioportal.bioontology.org/ontologies/GAZ.

Consortium, The UniProt. "UniProt: a hub for protein information ." *Oxford Journals* 43, no. D1 (2014).

DisGeNet. 10 2014. http://www.disgenet.org/web/DisGeNET/v2.1/dbinfo.

Goh, Kwang-Il, Michael Cusick, David Valle, Barton Childs, Marc Vidal, and Albert-László Barabási. "The Human Disease Network." *Proc Natl Acad Sci USA*, 4 2007.

Google Inc; Microsoft Inc; Yahoo Inc. 2014. http://schema.org/.

Heath, Tom, and Christian Bizer. *Linked Data: Evolving the Web into a Global Data Space.* 1. Berlin: Morgan & Claypool, 2011.

Kanehisa, M, S Goto, Y Sato, M Kawashima, M Furumichi, and M Tanabe. "Data, information, knowledge and principle: back to metabolism in KEGG." *Nucleic Acids Res*, Jan 2014.

Kuhn, M, M Campillos, I Letunic, LJ Jensen, and P Bork. "A side effect resource to capture phenotypic effects of drugs." *Epub* (NCBI), 1 2010.

Law, V, et al. "DrugBank 4.0: Shedding new light on drug metabolism." *PubMed*, no. 24203711 (2014).

National Institute of Allergy and Infectious Diseases; National Institutes of Health; Department of Health and Human Services. 2014. https://www.vectorbase.org.

Stanford University. 2014. https://www.pharmgkb.org/.

United States National Library of Medicine. 10 1, 2014. http://dailymed.nlm.nih.gov/.

W3C OWL Working Group. 2012. http://www.w3.org/TR/2012/REC-owl2-overview-20121211/.

W3C RDF Working Group. 2004. http://www.w3.org/TR/2004/REC-rdf-mt-20040210/.

W3C SPARQL Working Group. "SPARQL 1.1 Query Language." *W3C Recommender*. March 2013. http://www.w3.org/TR/sparql11-query/ (accessed October 2014).

Wikimedia Foundation. 2014. http://www.wikidata.org

Scaffolding the Mitochondrial Disease Ontology from extant knowledge sources

Jennifer D. Warrender and Phillip Lord*

School of Computing Science, Newcastle University, Newcastle-upon-Tyne, UK

ABSTRACT

Bio-medical ontologies can contain a large number of concepts. Often many of these concepts are very similar to each other, and similar or identical to concepts found in other bio-medical databases. This presents both a challenge and opportunity: maintaining many similar concepts is tedious and fastidious work, which could be substantially reduced if the data could be derived from pre-existing knowledge sources. In this paper, we describe how we have achieved this for an ontology of the mitochondria using our novel ontology development environment, the Tawny-OWL library.

1 INTRODUCTION

Bio-medical ontologies vary in size, with largest containing millions of concepts. Building ontologies of this size is complex, time-consuming and expensive and just as challenging to maintain and update.

Ontologies are only one of many mechanisms for the computational representation of knowledge. In some cases, ontologies are created where many of the needed concepts will be available elsewhere as terms in different structured representations. Being able to reuse these representations as a *scaffold* for the rest of an ontology might be able to reduce the cost and work-load of producing ontologies.

This is evidenced by, for instance, SIO (Dumontier *et al.*, 2014) which contains a list of all the chemical elements. Or the Gene Ontology (GO) (Ashburner *et al.*, 2000), which contains many terms related to chemical homeostasis, each of which need to relate to a specific chemical described in ChEBI (Hastings *et al.*, 2013). In addition to being described elsewhere, these concepts are often highly similar to each other. In extreme cases such as the amino acid ontology (Stevens and Lord, 2012), ontologies can consist of only related concepts, and "support" concepts that are used to describe them.

One solution to this is the use of patterns. A pattern is an abstract specification of an ontology axiomatisation with a number of "variables". The pattern is instantiated by providing values for these variables, which are then expanded into the full axiomatisation providing one or more concepts.

Patterns have been implemented by a number of different tools, which differ in how the patterns are specified, and how and when the values are provided for the variables. For example, *termgenie* is a website which allows submission to GO (and others) (Dietze *et al.*, 2014). Variable values are entered through a form which then generates axioms, definitions and cross-references. For instance, this is the axiomatisation from termgenie when defining the term "cytosine homeostasis"

```
is_a: GO:0048878 {is_inferred="true"}
  ! chemical homeostasis
intersection_of: GO:0048878
  ! chemical homeostasis
intersection_of:
  regulates_levels_of CHEBI:16040 ! cytosine
relationship:
  regulates_levels_of CHEBI:16040
  {is_inferred="true"} ! cytosine
```

As well as the axiomatisation, termgenie also generates a number of different annotations including a definition, submitter information, and status. With termgenie, patterns are specified through the use of JavaScript functions.

In addition to termgenie, other systems also allow patterns. For example, both the desktop and web version of Protégé contain forms, which grant users the ability to customise the GUI and specify several axioms at once. In this case, patterns are declaratively defined (implicitly, with a GUI design) in XML (Tudorache *et al.*, 2013). Applications like Populous (Jupp *et al.*, 2011) and Rightfield (Wolstencroft *et al.*, 2011) use spreadsheets or spreadsheet-like interfaces to enter data, which is then transformed into a set of OWL axioms based on a pattern. In the case of these two, the patterns are specified in OPPL, a pattern language for OWL which can also be used independently (Egana Aranguren *et al.*, 2009). Finally, the Brain API allows programmatic construction of ontologies in an easy to use manner using Java (Croset *et al.*, 2013).

While these systems are all aimed at somewhat different use-cases, they all address the same problem; how to produce a large number of concepts all of which are similar, and to do so with a high-degree of repeatability. However, the use of this form of patternised ontology tool presents a number of problems. These tools provide a mechanism for adding many axioms at once, but not removing them again[1]. If the knowledge changes, then this is a problem as the axioms added from a given pattern need to be removed or updated. Furthermore, if the knowledge engineering changes i.e. the pattern is updated, then all axioms added from any use of the pattern must also be updated.

In this paper, we describe how we have addressed these problems with the Mitochondrial Disease Ontology (MDO), through the use of the Tawny-OWL environment, which is a fully programmatic environment for ontology development. With Tawny-OWL, we can use a *pattern-first* ontology development process, building with patterns and data from extant knowledge sources from the start. This has allowed us to generate a *scaffold* which we can then populate further with hand-crafted links between parts of this scaffold where the knowledge exists. As a result, it is possible to

*To whom correspondence should be addressed: phillip.lord@newcastle.ac.uk

[1] OPPL can remove axioms as well as add them but this is not automatic.

update both the knowledge and the patterns by simply regenerating the ontology. This process promises to aid in both the construction and maintenance of ontologies.

The MDO is available from `https://github.com/jaydchan/tawny-mitochondria`. Tawny-OWL is available from `https://github.com/phillord/tawny-owl`.

2 THE MITOCHONDRIA DISEASE ONTOLOGY (MDO)

Mitochondria are complex organelles found in most eukaryotic cells. Their key function is to enable the production of ATP through oxidative phosphorylation, providing usable energy for the rest of the cell. The mitochondria carry their own small genome containing 37 genes in human. Many other genes are involved in producing proteins involved in mitochondrial function, but these are encoded in the nuclear genome. A number of mitochondrial genes are associated with diseases; the first identified of these is the MELAS (Pavlakis *et al.*, 1984), which is most commonly caused by a point mutation in a tRNA found in the mitochondrial genome.

As with many areas of biology, mitochondrial research is a large, knowledge-rich discipline. Our purpose with the MDO is to attempt to formalise this knowledge, using an incremental or "pay-as-you-go" data integration approach. The ontology here serves as a tool for reasoning and knowledge exploration, rather than to form as a reference ontology (Stevens and Lord, 2008). This is an approach we have previously found useful in classifying phosphatases (Wolstencroft *et al.*, 2006). The hope is that we can incorporate new knowledge as it is released, checking it for consistency and cross-linking it with existing knowledge.

3 TAWNY-OWL

In this section, we give a brief description of Tawny-OWL (Lord, 2013) and how it supports pattern-first development. Tawny-OWL is a library written in Clojure, a dialect of lisp. It wraps the OWL API (Horridge and Bechhofer, 2011) and allows the fully programmatic constructions of ontologies. It has a simple syntax which was modelled on the Manchester Syntax (Horridge and Patel-Schneider, 2012), modified to integrate well with Clojure. It can be used to make simple statements in OWL:

```
(defclass A :super (some r B))
```

which makes defines a new class A such that $A \sqsubseteq \exists\ r\ B$. Although this is similar to the equivalent Manchester Syntax statements, Tawny-OWL provides a feature called "broadcasting" which is, essentially a form of pattern. So this following statement:

```
(some r B C)
```

is equivalent to the two statements $\exists\ r\ B$ and $\exists\ r\ C$. We apply the first two arguments (`some` and `r`) to the remaining ones consecutively. It also provides simple patterns, such as the covering axiom, so:

```
(some-only r B C)
```

is equivalent to three statements $\exists\ r\ B$, $\exists\ r\ C$ and $\forall\ r\ (B \sqcup C)$. While the patterns shown here are provided by Tawny-OWL, end ontology developers are using the same programmatic environment. Patterns are encoded as functions and instantiated with function calls. For instance, we could define `some-only` as follows:

```
(defn some-only [property & classes]
 (list (some property classes)
       (only property
             (or classes))))
```

Here `defn` introduces a new function, `property & classes` are the arguments, and `list` packages the return values as a list. `some`, `only` and `or` are defined by Tawny-OWL as the appropriate OWL class constructors.

It is, therefore, possible to build *localised patterns* — custom patterns for use predominately with the current ontology (Warrender, 2015). Patterns can call each other and can be of arbitrary complexity. The use of Tawny-OWL, therefore, inverts the usual style of ontology development. Non-patternised classes are just trivial instantiations of patterns.

4 BUILDING A MITOCHONDRIAL SCAFFOLD

Following a requirements gathering phase for MDO, it was clear from our competency questions (for example "What are all the genes/proteins that are associated with a specific syndrome?") that we needed many concepts which were heavily repetitive, and further which have comprehensive and curated lists available. We describe these parts of the domain knowledge as the *scaffold*. For example, there are around 761 genes whose products are involved in mitochondrial function. Classes representing these genes do not, in the first instance, require complex descriptions, and are defined within MDO as follows:

```
(defclass Gene)

(defn gene-class [name]
 (owl-class name :label name :super Gene))
```

This pattern is then populated using a simple text file, with the 761 gene names present. The gene pattern is an extremely simple pattern, as these concepts are self-standing. Other parts of the ontology are even simpler; for instance, for describing mitochondrial anatomy, the classes have similar complexity to the genes, but there are only 15. In this case, classes are defined with a pattern and a list "hard-coded" into the MDO source code, rather than using an external text file. Other patterns are more complex. For instance, the subclasses of `Disease` are defined as follows:

```
(defn disease-class [name omim lname]
 (let [disease
       (owl-class name
                  :label name
                  :super Disease)]
  (if-not (nil? omim)
   (refine disease
           :annotation
           (see-also
            (str "OMIMID:" omim))))
  (if-not (nil? lname)
   (refine disease
           :label
           (str "Long name:" lname)))))
```

This function adds two annotations to each disease class, if they are available. This function also demonstrates the use of conditionals (`if`), predicates (`nil?`) and string concatenation (`str`); these are not provided by Tawny-OWL, but by Clojure and demonstrate the value of building Tawny-OWL inside a fully programmatic environment.

5 FITTING OUT THE SCAFFOLD

The top-level of the MDO is shown in Figure 1. Of these classes, "Paper" and "Term" are described later.

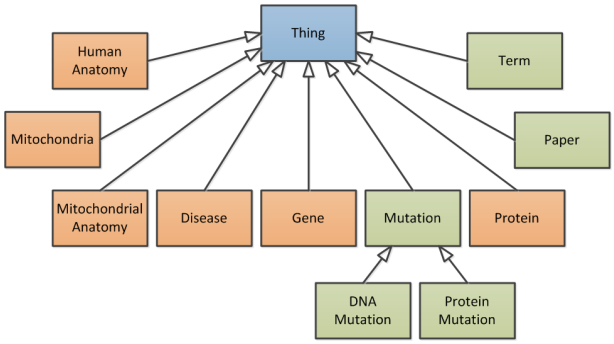


Fig. 1. The top-level structure of Mitochondrial Disease Ontology. Classes that are a part of the scaffold are coloured in orange, while classes that are built on top of the scaffold are coloured in green.

The remaining classes define the scaffold, which now has a total of 1357 classes; a break-down of these classes and their sources is shown in Table 1.

Class type	Count	Data source
Disease	41	UMDF website
Gene	761	The NCBI Gene portal
Human Anatomy	61	The Terminologia Anatomica.
Mitochondrial Anatomy	15	Mitochondrial Research Group
Protein	479	UniProt

Table 1. Table showing the type, number of and data source for each generic mitochondrial ontology class

For the next stage of the process, we are now building on top of this scaffold, using hand-crafted and bespoke knowledge. This is being achieved by manual extraction of knowledge from papers about mitochondria. Our initial process is to find references in papers to the terms that are represented by classes we have built in the scaffold, and draw explicit relationships between these papers and the scaffolded knowledge that they describe. Currently, these classes also use a patternised approach; the raw data is held in a bespoke (but human readable) syntax[2], which is then parsed and used to instantiate patterns. In total, there are now 2174 classes created from this approach from around 30 papers. These terms currently are not defined beyond their name and the source paper from which they were identified. We do not consider them directly as part of the scaffold, as they are not from an extant knowledge source, but one that we have created; they are the first layer build on top of our scaffold. We expect future layers to use the Tawny-OWL syntax directly, as the knowledge increases in complexity and decreases in regularity.

6 RESILIANCE TO CHANGE

One key feature of our development process is that the OWL which defines the MDO is no longer *source code* but generated. Rather it is generated from patterns defined in Tawny-OWL and text files which are used to instantiate these patterns. The in-memory OWL classes and associated OWL files are generated on-demand, by *evaluating* the patterns. Effectively, we regenerate the ontology every time we restart the environment. In this section, we consider the types of changes that can happen, and how these changes impact on MDO.

The scaffold of MDO is sensitive to changes in its dependency knowledge sources. First, new terms can be entered into extant sources, which will necessitate the addition of new classes. For the MDO, this simply necessitates re-importing the knowledge. The addition of equivalent new classes will then happen automatically according to the patterns already defined; no other changes should be necessary for the MDO, although we may wish to refer to the new classes in other parts of the ontology.

Second, terms may be removed from dependencies; so, for example, a disease may be redefined by the UMDF. In many cases, for the MDO, this is not problematic – the equivalent classes will simply disappear from the ontology. Tawny-OWL provides two features to help with changes to terms in the scaffold when these terms are also referred to outside of the scaffold. Tawny-OWL uses a "declare-before-use" semantics, so removal of classes from the scaffold will cause fail-fast behaviour when they are used elsewhere. The Brain environment uses the same semantics for similar reasons (Croset *et al.*, 2013). In addition, Tawny-OWL provides a "deprecation" facility which allows the developer to continue refer to terms from the scaffold which have been removed, but to receive warnings about this use; this is rather like obsolescence, but happens automatically[3].

Third, the MDO scaffold can also cope straight-forwardly with changes to patterns. As with the addition or removal of terms from dependencies, pattern changes will simply take place by re-evaluating the ontology.

Finally, the MDO is resilient to changes in ontology engineering conventions. For example, MDO does not use OBO style numeric identifiers, nor provide stable IRIs for integration with linked data sources since these are not critical at the current time[4]. They, however, could be added easily to all existing (and future) terms in a few lines of code, using an existing facility within Tawny-OWL for minting and persisting numeric identifiers in an automatic, yet managed, way. This change would just alter IRIs and would have

[2] In this case *EDN* which is a text representation of Clojure data structures; it looks rather like JSON.

[3] Tawny-OWL is implemented in a Lisp and so is homoiconic; this makes it particularly straight-forward to automate code updates if we choose.

[4] Our initial intention was to use PURLS from `www.purl.org` but have found practical problems with generating these.

no impact on references between concepts inside or outside of the scaffold.

In conclusion, as well as enabling rapid construction of the MDO, we believe that the pattern-first scaffolding approach should also allow easy maintenance of the ontology.

7 DISCUSSION

In this paper, we have described how we have used a number of extant knowledge sources, combined with patterns defined using the Tawny-OWL library to rapidly, reliably and repeatedly construct a scaffold for MDO.

We have previously used a related patternised methodology to construct a complex ontology describing human chromosome rearrangements (i.e. The Karyotype Ontology (KO) (Warrender and Lord, 2013b)). However, unlike KO, the mitochondrial knowledge we want to encapsulate is found in numerous independent sources (e.g. published papers and online databases) and in a variety of formats (e.g. "free text" and CSV); the use of several patterns to form a scaffold is unique to MDO. Conversely, the axiomatisation of MDO from these sources is simple; this cannot be said for KO, most of which is generated from a single large pattern (Warrender and Lord, 2013a). In addition, while our knowledge of the karyotype is constrained and is essentially finished, the community's understanding of mitochondria and mitochondrial disease is incomplete and will grow in response to the demands of changing knowledge.

This methodology is extremely attractive for a number of reasons. First of all, it allows a very rapid way of scaffolding an ontology for a complex area of knowledge. At this stage, most of the classes created are simple and self-standing, although in some cases do have relationships to other entities in the scaffold. At this point, we have built the ontological equivalent of a data warehouse: terms have been taken from elsewhere and have undergone a form of schema reconciliation into ontological classes.

One key feature of the MDO is that it has been built using tools designed for software development; these tools are relatively advanced and well-maintained[5] (Lord, 2013). Moreover, recreating the MDO ontology from our original Tawny-OWL source code is an intrinsic part of the development process; there is no complex release process and any ontology developer can recreate the OWL file with a single command. While, the system as it stands has a high-degree of replicability, the design decisions implicit in the source code are not necessarily apparent. For the basic scaffold this is, perhaps, not a major issue, however as MDO is developed outside of its scaffold , we expect to integrate more documentation into the source code itself, using *lentic*, a recently developed tool for literate programming (Lord, 2015).

We believe that the engineering process that we have used to build the scaffold is resilient to change, as described in Section 6. Despite this resilience, our use of external sources of knowledge does bring with it new dependencies, with all the issues that this entails for change management. We believe that we can manage this by borrowing best practice from software engineering. Importing knowledge into the scaffold can, in many cases, happens entirely automatically from our extant knowledge sources. Considering just the gene lists, we can either import from a local, fixed copy of this list, or take the current version live from the NCBI portal. In software engineering terms, the former is a *release dependency* and provides stability, while the latter is a *snapshot dependency* which will fail-fast, allowing rapid incorporation of new knowledge. The latter is particularly useful within a continuous integration environment which are used with other ontologies (Mungall *et al.*, 2012), and are also fully supported by Tawny-OWL (Lord, 2013).

Although we have not described its usage here, with the MDO we are not forced to use Tawny-OWL for all development. It would be possible to combine predominately hand-crafted development using Protégé, for instance, with some patternised classes; for example, the OBI uses this approach (Brinkman *et al.*, 2010). For, the MDO, in fact almost all terms other than the top-level has been created from other syntaxes, generally a flat-file. For larger projects, we envisage that most ontology developers would not need to use the programmatic nature of Tawny-OWL. While we appreciate the value of a single environment, a tool should not force all users into it.

In this paper, we have described our approach to building the MDO using a patternised scaffold based around existing knowledge sources. While the work described in this paper allows us to integrate structured data into an ontology, we are now investigating new ways of integrating unstructured literate-based knowledge into our ontology; while we have started the process of formalising, this new knowledge is far from finished. As described in this paper, though, a pattern-first, scaffolded approach to ontology development has enabled us to make significant advances with the MDO. We believe that this approach is likely to be applicable to many other domains also.

[5] And, usefully, not dependent on academic developers for future maintenance.

REFERENCES

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature Genetics*, **25**(1), 25–29.

Brinkman, R., Courtot, M., Derom, D., Fostel, J., He, Y., Lord, P., Malone, J., Parkinson, H., Peters, B., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Soldatova, L., Stoeckert, C., Turner, J., Zheng, J., and the OBI consortium (2010). Modeling biomedical experimental processes with obi. *Journal of Biomedical Semantics*, **1**(Suppl 1), S7.

Croset, S., Overington, J. P., and Rebholz-Schuhmann, D. (2013). Brain: biomedical knowledge manipulation. *Bioinformatics*, **29**(9), 1238–1239.

Dietze, H., Berardini, T. Z., Foulger, R. E., Hill, D. P., Lomax, J., OsumiSutherland, D., Roncaglia, P., and Mungall, C. J. (2014). Termgenie - a web application for pattern-based ontology class generation. *Journal of Biomedical Semantics*, **5**(1), 48.

Dumontier, M., Baker, C., Baran, J., Callahan, A., Chepelev, L., Toledo, J. C., Del Rio, N., Duck, G., Furlong, L., Keath, N., Klassen, D., McCusker, J., Rosinach, N. Q., Samwald, M., Rosales, N. V., Wilkinson, M., and Hoehndorf, R. (2014). The Semanticscience Integrated Ontology (SIO) for biomedical research and knowledge discovery. *Journal of Biomedical Semantics*, **5**(1), 14+.

Egana Aranguren, M., Stevens, R., and Antezana, E. (2009). Transforming the axiomisation of ontologies: The ontology pre-processor language. *Nature Precedings*.

Hastings, J., de Matos, P., Dekker, A., Ennis, M., Harsha, B., Kale, N., Muthukrishnan, V., Owen, G., Turner, S., Williams, M., and Steinbeck, C. (2013). The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013. *Nucleic Acids Research*, **41**(D1), D456–D463.

Horridge, M. and Bechhofer, S. (2011). The OWL API: A Java API for OWL ontologies. *Semant. web*, **2**(1), 11–21.

Horridge, M. and Patel-Schneider, P. F. (2012). Owl 2 web ontology language manchester syntax (second edition). Technical report.

Jupp, S., Horridge, M., Iannone, L., Klein, J., Owen, S., Schanstra, J., Wolstencroft, K., and Stevens, R. (2011). Populous: a tool for building owl ontologies from templates. *BMC Bioinformatics*, **13**(Suppl 1), S5.

Lord, P. (2013). The Semantic Web takes Wing: Programming Ontologies with Tawny-OWL. http://arxiv.org/abs/1303.0213.

Lord, P. (2015). Lenticular text: Looking at code from different angles. http://www.russet.org.uk/blog/3035.

Mungall, C., Dietze, H., Carbon, S., Ireland, A., Bauer, S., and Lewis, S. (2012). Continuous integration of open biological ontology libraries. http://bio-ontologies.knowledgeblog.org/405.

Pavlakis, S. G., Phillips, P. C., DiMauro, S., De Vivo, D. C., and Rowland, L. P. (1984). Mitochondrial myopathy, encephalopathy, lactic acidosis, and strokelike episodes: a distinctive clinical syndrome. *Ann. Neurol.*, **16**(4), 481–488.

Stevens, R. and Lord, P. (2008). Application of ontologies in bioinformatics. In S. Staab and R. Studer, editors, *Handbook on Ontologies in Information Systems*. Springer, second edition.

Stevens, R. and Lord, P. (2012). Semantic publishing of knowledge about amino acids. http://ceur-ws.org/Vol-903/paper-06.pdf.

Tudorache, T., Nyulas, C., Noy, N., and Musen, M. (2013). Using semantic web in icd-11: Three years down the road. In H. Alani, L. Kagal, A. Fokoue, P. Groth, C. Biemann, J. Parreira, L. Aroyo, N. Noy, C. Welty, and K. Janowicz, editors, *The Semantic Web ISWC 2013*, volume 8219 of *Lecture Notes in Computer Science*, pages 195–211. Springer Berlin Heidelberg.

Warrender, J. D. (2015). *The Consistent Representation of Scientific Knowledge: Investigations into the Ontology of Karyotypes and Mitochondria.* Ph.D. thesis, School of Computing Science, Newcastle University.

Warrender, J. D. and Lord, P. (2013a). A pattern-driven approach to biomedical ontology engineering. *SWAT4LS 2013*.

Warrender, J. D. and Lord, P. (2013b). The Karyotype Ontology: a computational representation for human cytogenetic patterns. *Bio-Ontologies SIG 2013*.

Wolstencroft, K., Lord, P., Tabernero, L., Brass, A., and Stevens, R. (2006). Protein classification using ontology classification. *Bioinformatics*, **22**(14), e530–538.

Wolstencroft, K., Owen, S., Horridge, M., Krebs, O., Mueller, W., Snoep, J. L., du Preez, F., and Goble, C. (2011). RightField: embedding ontology annotation in spreadsheets. *Bioinformatics*, **27**(14), 2021–2022.

Analysis of the evolution of ontologies using OQuaRE: Application to EDAM

Manuel Quesada-Martínez, Astrid Duque-Ramos, Jesualdo Tomás Fernández-Breis*
Facultad de Informática, Universidad de Murcia, IMIB-Arrixaca, CP 30100 Murcia

ABSTRACT

In recent years, the biomedical community has developed a significant number of ontologies. The curation of biomedical ontologies is a complex task, which has the practical implication of a high number of versions of ontologies in short time, because biomedical ontologies evolve rapidly. New versions are periodically published in ontology repositories. Ontology designers need to be supported for the effective management of the evolution of biomedical ontologies given this level of activity, because the different changes may affect the engineering and quality of the ontology. This is why we think that there is a need for methods that contribute to the analysis of the effects of changes and evolution of ontologies.

In this paper we approach this issue from the ontology quality perspective. In previous works we have developed an ontology evaluation framework based on quantitative metrics, called OQuaRE. Here, OQuaRE will be used as a core component in a method that permits to analyze the different versions of biomedical ontologies using a common framework. The objective is to help ontology developers to study the evolution of ontology versions in terms of changes in the quality dimensions analyzed in OQuaRE. In this work we explain how OQuaRE can be adapted for supporting this process and report the application of the method to 16 versions of the EDAM ontology. Discussion is provided on the evolution of the quality scores of those versions according to the OQuaRE quality perspective.

1 INTRODUCTION

In recent years, the biomedical community has increased its effort in the development of good ontologies and this will continue in the future (Hoehndorf *et al.*, 2014). As a consequence, ontology developers publish their new ontologies across the Internet, and they are accessible from different sources. BioPortal (Whetzel *et al.*, 2011) contains 428 ontologies at the time of writing, and new content is published every week. BioPortal provides for automatic updates by user submissions of new versions, which are accessible via web browsers and through web services (Whetzel *et al.*, 2011).

The curation of ontologies is a complex task because of their high level of activity and rapid evolution (Malone *et al.*, 2010). For this reason, their number and versions grow rapidly. The analysis of versions was introduced by Klein and Fensel (2001), who defined ontology versioning as the ability to handle changes in ontologies by creating and managing different variants of it and pointed out the importance of highlighting differences between versions. Later, Noy *et al.* (2003) claimed that a versioning system for ontologies must compare and present structural changes rather than changes in text representation or source files. They described a version-comparison algorithm that produces a structural difference between ontologies, which were presented to users through an interface for analysing them (Noy *et al.*, 2004). Later, Malone *et al.* (2010) presented Bubastis that reports on 5 major types of ontology changes: added or removed axioms to an existing named class (NC), NCs added, NCs made obsolete and edited annotation properties. Bubastis[1] was used in (Malone *et al.*, 2010) for measuring the level of activity of bio-ontologies, and this is used in BioPortal to generate reports about changes between 2 consecutive versions. Recently, Copeland *et al.* (2013) focused on changes in asserted and inferred axioms taking into account reasoning capabilities in ontologies (Wang *et al.*, 2004).

In this work, we are interested in studying the evolution of ontologies from the perspective of ontology quality. The analysis of quality in ontologies has been addressed in different ways in the ontology evaluation community (Gangemi *et al.*, 2006; Tartir and Arpinar, 2007; Ma *et al.*, 2009; Duque-Ramos *et al.*, 2011). Gangemi *et al.* (2006) approached it as a diagnostic task based on ontology descriptions, using three categories of criteria (structural, functional and usability profiling). Similarly, Rogers *et al.* (2006) proposed an approach using four qualitative criteria (philosophical rigour, ontological commitment, content correctness, and fitness for a purpose). Quantitatively, Yao *et al.* (2005) and Tartir and Arpinar (2007) presented metrics for evaluating structural properties in the ontology. Recently, Duque-Ramos *et al.* (2011) adapted the SQuaRE standard for software quality evaluation for defining a qualitative and quantitative ontology quality model.

In this paper, we propose a method that combines ideas from the ontology evaluation and ontology versioning field by adapting the OQuaRE methods for the needs of the study of changes between versions of ontologies. In this paper, we will explain the method and we will exemplify its application to the study of 16 versions of the ontology of Bioinformatics operations, types of data, formats, and topics (EDAM)[2]. The analysis of the results will permit to detect the impact of the series of changes in the ontology on the quality measurements offered by OQuaRE, which may contribute to learn about the engineering of the EDAM ontology. Our results will be compared with the ones obtained with Bubastis to study the relations between the level of activity of ontologies and changes in the OQuaRE quality scores. We believe this kind of method may contribute to generate new insights about biomedical ontologies.

2 METHODS

2.1 OQuaRE

OQuaRE (Duque-Ramos *et al.*, 2011) is an ontology quality evaluation framework based on the software product quality SQuaRE. OQuaRE aims at defining all the elements required for ontology evaluation: evaluation support, evaluation process and

*To whom correspondence should be addressed: jfernand@um.es

[1] http://www.ebi.ac.uk/efo/bubastis/

[2] http://edamontology.org

Characteristic	Description	Associated subcharacteristics
Structural	Formal and semantic relevant ontological properties that account for: the correct use of formal properties, clarity of cognitive distinctions and appropriate use of ontology modelling primitives and principles	"formalisation", "formal relations support", "redundancy", "consistency", "tangledness", "cohesion"
Functional Adequacy	Capability of the ontologies to be deployed fulfilling functional requirements, that is, the appropriateness for its intended purpose according to state-of-theart literature Stevens *et al.* (2008)	"reference ontology", "controlled vocabulary", "schema and value reconciliation", "consistent search and query", "knowledge acquisition", "clustering and similarity", "indexing and linking", "results representation", "text analysis", "guidance and decision trees" and "knowledge reuse and inferencing"
Reliability	Capability of an ontology to maintain its level of performance under stated conditions for a given period of time	"recoverability" and "availability"
Operability	Effort needed for the ontology use. Individual assessment of such use, by a stated or implied set of users	"learnability"
Compatibility	Capability of two or more ontologies to exchange information and/or to perform their required functions while sharing a hw/sw environment	"replaceability"
Maintainability	Capability of ontologies to be modified for changes in environments, in requirements or in functional specifications	"modularity", "reusability", "analysability", "changeability", "modification stability" and "testability"
Transferability	Degree to which the ontology can be transferred from one environment (e.g., operating system) to another	"adaptability"

Table 1. OQuaRE characteristics and subcharacteristics used in our method

metrics. The main objective of OQuaRE is to provide an objective, standardized framework for ontology quality evaluation, which could be applied in a number of situations.

OQuaRE is structured in 3 levels: quality characteristics, subcharacteristics and metrics. The evaluation of an ontology comprises a score for each quality characteristic, which depends on the evaluation of the its associated subcharacteristics. Similarly, the evaluation of a particular subcharacteristic depends on its associated metrics.

Table 1 describes the OQuaRE characteristics and subcharacteristics we use in this work. OQuaRE metrics adapt successful metrics from both ontology and software engineering communities (Tartir *et al.*, 2005; Yao *et al.*, 2005), which we briefly describe in Table 2. The complete specification of the OQuaRE quality model, including the associations between subcharacteristics and metrics, can be found at http://miuras.inf.um.es/oquarewiki.

OQuaRE metrics generate quantitative values in different ranges, so they are scaled into the range 1 to 5, which is the scale used in SQuaRE based approaches. There, 1 means not acceptable, 3 is minimally acceptable, and 5 is exceeds the requirements. The scaling method is based on the recommendations and best practices of the Software Engineering community for software metrics and ontology evaluation metrics (see the scale at the OQuaRE website).

OQuaRE provides a flexible analysis framework because ontologies can be analysed at different granularity levels: metric, subcharacteristic, characteristic and globally.

2.2 The method

Fig. 1 shows different stages of our method. We propose a method focused on measuring changes between different versions of the same ontology in terms of its global quality.

DEFINITION 1. ***Versioned corpus of an ontology*** *(vC): vC is a list of versions $[\theta_{v1}, \theta_{v1+1}..., \theta_{vt}]$ of the same ontology θ, where a time criterion must sequentially order vC.*

Ontologies can be found in different sources and formats, so we propose to normalise vC before applying OQuaRE. In the normalisation, we check that they are consistent, remove deprecated classes and save them in a the same OWL format.

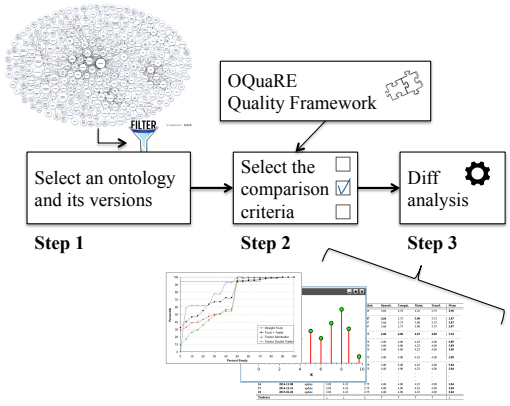


Fig. 1. Three main stages of the method for evaluating how to evolve the OQuaRE quality scores in a versioned corpus (vC) of an ontology

The second step permits to select which OQuaRE quality criteria are used to analyse the evolution of the ontologies. We define a comparison criterion as follows:

DEFINITION 2. ***Comparison criterion*** *($f(\theta)$): $f(\theta)$ is a quantifiable score that measures the capability of the ontology $\theta_{vi} \in vC$ to fulfill some criterion.*

When we compare different versions of the same ontology differences should be highlighted (Noy *et al.*, 2003). In our context, given an ontology θ and two consecutive versions $\theta_{vi}, \theta_{v(i+1)} \in vC$:

DEFINITION 3. ***Change:*** *there is a change if $f(\theta_{vi}) \neq f(\theta_{v(i+1)})$, being $f(\theta)$ a comparison criterion.*

For example, if we obtain average scores of θ_{v1}=3.75 and θ_{v2}=3.87 for different versions, then the change of 0.12 indicates that the second one has a higher score than the first one. Provided

OQuaRE metric	Description
ANOnto	mean number of annotation properties per class
AROnto	number of restrictions of the ontology per classes
CBOnto	number of superclasses divided by the number of class minus the subclasses of Thing
CROnto	mean number of individuals per class
DITOnto	length of the largest path from Thing to a leaf class
INROnto	mean number of subclasses per class
NACOnto	mean number of superclasses per leaf class
NOCOnto	mean number of the direct subclasses per class minus the subclasses of Thing
NOMOnto	mean number of object and data property usages per class
LCOMOnto	mean length of all the paths from leaf classes to Thing
RFCOnto	number of usages of object and data properties and superclasses divided by the number of classes minus the subclasses of Thing
RROnto	number of usages of object and data properties divided by the number of subclassof relationships and properties
TMOnto	mean number of classes with more than 1 direct ancestor
WMCOnto	mean number of properties and relationships per class

Table 2. OQuaRE metrics and a brief description of how we calculate them

that OQuaRE metrics are scaled into the range 1-5 we need to differentiate which changes generate a variation in the scale of the metric score.

DEFINITION 4. ***Change in scale:*** *it is a sort of change where $f(\theta_{vi})$ and $f(\theta_{v(i+1)})$ are associated with different levels in the scale 1-5.*

In our example, there is no change in scale for θ_{v1} and θ_{v2}. However, changes from 3.25 to 2.87 or from 4.10 to 3.98 would make a change in the scale.

3 RESULTS AND DISCUSSION

We apply our method to the EDAM ontology. EDAM is an ontology of well established and familiar concepts that are prevalent within bioinformatics. EDAM includes types of data, data identifiers, data formats, operations and topics. We have choosen this ontology as exemplar because:

- It is well documented and its developers use a control version system[3] (CVS) so that we can trace changes.
- Its source files are accessible online. The latest version (v1.9) is published in the official project web page. Links to old versions can be found in BioPortal (18) and in the CVS (13).
- It has received 775 mean visits per month since October 2013 and 5 declared projects use EDAM, so it is a relevant ontology.
- Its number of versions and size (2 597 classes on average) makes its appropriate for this initial study.

We configured the experiment as follows. The *versioned corpus* is composed by the 18 EDAM versions in BioPortal as CVS content. We performed the *diff analysis* using OQuaRE metrics, subcharacteristics and characteristics as *comparison criteria*. We automatically processed the *versioned corpus* using a home-made software tool that implements the methods described in the previous section. This tool uses the OWL API[4] and Neo4j (http://neo4j.com) (paths metrics) for the calculation of OQuaRE metrics. 4 out of 18 versions were discarded by the tool: one could not be processed by the OWL API, and the other three were found inconsistent by Hermit (http://hermit-reasoner.com). In order to study the impact of deprecated classes in the results, we performed two studies: one with the ontologies containing the deprecated classes and one removing them. After this removal, v.13 and v.14 became consistent, so they were processed and included in the second study. Table 3 shows the results obtained in the characteristics level for the 16 versions in both studies. The whole set of results is available at [5], which includes scores and other information in the subcharacteristics and metrics levels.

3.1 Changes in Quality Scores

According to Table 3 the mean quality score ranges from 3.99 in the first version to 3.85 in the last one. The changes in this score do not generate a change in the scale. In fact, the EDAM ontology has always stayed between 3 and 4. Taking into account the OQuaRE scale, a 3-upper score reveals that good ontological principles seem to have been applied by the EDAM developers. In order to get insights about the engineering and evolution of the ontology, we continue by analysing changes in scale identified for different quality characteristics (see numbers in bold in Table 3).

3.1.1 Increase in quality scores: the Operability, Compatibility, Maintainability and Transferability characteristics increased from level 3 to 4 between v.4 and 5. Moreover, the ontology has maintained the score at this level since then. This behaviour happens for all the associated subcharacteristics. These scores are not included in the paper due to space constraints, but can be found in the result webpage. Descriptively, the highest score is found for Maintainability. The scores for its subcharacteristics "Reusability", "Analisability", "Changeability", "Modification stability" and "Testability", qualitatively make the ontology more reusable, and reduces negative side-effects due to changes in the ontology. In addition to this, these scores mean that it is easier to validate and detect flaws in EDAM. A similar reasoning can be done for the other three characteristics using the information in Table 1.

The OQuaRE metrics level reveals more information about the ontology components. In this level, the score for 9 OQuaRE metrics did not change for any version. The ones that changed are shown in Fig. 2. NOMOnto and RFCOnto are responsible for the increase from 4 to 5. The decrease in NOMOnto means that the mean number of property usage per class is lower, which is good in terms of maintainability of the ontology. RFCOnto is related to the usage of properties too.

3.1.2 Decrease in quality scores : the "Reliability" characteristic decreases from 3 to 2 between v.1 and v.2, whereas and the "Structural" characteristic does it from 4 to 3 between v.10 and v.11. The lowest score for the "Structural" characteristic is for Cohesion, which is related to the LCOMOnto metric (see Fig. 2) that uses the number of paths in the ontology in its calculation.

[3] https://github.com/edamontology/edamontology/releases

[4] http://owlapi.sourceforge.net

[5] http://miuras.inf.um.es/oquare/icbo2015

Version	Date	Status	Struct.		F. Adeq.		Reliab.		Operab.		Compat.		Maint.		Transf.		Mean	
			Org.	Nrm.	Org.	Nrm.	Org.	Nrm.	Org.	Nrm.	Org.	Nrm.	Org.	Nrm.	Org.	Nrm.	Org.	Nrm.
1	**2010-05-14**	beta	4.67	4.67	4.61	4.61	3.25	3.25	3.83	3.83	3.75	3.75	4.10	4.10	3.75	3.75	3.99	3.99
2	**2010-05-28**	beta	4.50	4.50	4.60	4.60	**2.88**	**2.88**	3,67	3.67	3.75	3.75	3.99	3.99	3.75	3.75	3.88	3.88
3	**2010-08-18**	beta	4,50	4.50	4.60	4.60	2.88	2.88	3.67	3.67	3.75	3.75	3.99	3.99	3.75	3.75	3.88	3.88
4	**2010-10-07**	beta	4,50	4.50	4.60	4.60	2.88	2.88	3.67	3.67	3.75	3.75	3.99	3.99	3.75	3.75	3.88	3.88
5	**2010-12-01**	beta	4.17	4.17	4.46	4.46	2.75	2.75	**4.00**	**4.00**	**4.00**	**4.00**	**4.23**	**4.23**	**4.00**	**4.00**	3.94	3.94
6	**2011-01-22**	beta	4.00	4.00	4.28	4.28	2.75	2.75	4.00	4.00	4.00	4.00	4.23	4.23	4.00	4.00	3.90	3.90
7	**2011-06-17**	beta	4.00	4.00	4.28	4.28	2.75	2.75	4.00	4.00	4.00	4.00	4.23	4.23	4.00	4.00	3.90	3.90
8	**2011-12-05**	beta	4.00	**3.83**	4.28	4.27	2.75	2.38	4.00	**3.83**	4.00	4.00	4.23	4.12	4.00	4.00	3.90	3.78
10	**2012-12-10**	beta	4.00	3.83	4.28	4.27	2.75	2.38	4.00	3.83	4.00	4.00	4.23	4.12	4.00	4.00	3.90	3.78
11	**2012-12-14**	release	**3.83**	3.83	4.11	4.27	2.75	2.38	4.00	3.83	4.00	4.00	4.23	4.12	4.00	4.00	3.85	3.78
12	**2014-02-18**	update	3.83	3.83	4.11	4.27	2.75	2.38	4.00	3.83	4.00	4.00	4.23	4.12	4.00	4.00	3.85	3.78
13	**2014-09-26**	update	-	3.83	-	4.27	-	2.38	-	3.83	-	4.00	-	4.12	-	4.00	-	3.78
14	**2014-11-14**	update	-	**4.00**	-	4.28	-	2.75	-	**4.00**	-	4.00	-	4.23	-	4.00	-	3.90
16	**2014-12-08**	update	3.83	4.00	4.11	4.28	2.75	2.75	4.00	4.00	4.00	4.00	4.23	4.23	4.00	4.00	3.85	3.90
17	**2014-12-16**	update	3.83	**3.83**	4.11	4.11	2.75	2.75	4.00	4.00	4.00	4.00	4.23	4.23	4.00	4.00	3.85	3.85
18	**2015-02-02**	update	3.83	3.83	4.11	4.11	2.75	2.75	4.00	4.00	4.00	4.00	4.23	4.23	4.00	4.00	3.85	3.85

Table 3. OQuaRE characteristics metric values for two versions of the EDAM ontology. These values are scaled from 1 to 5, where 1 is not acceptable and 5 exceeds the requirements.

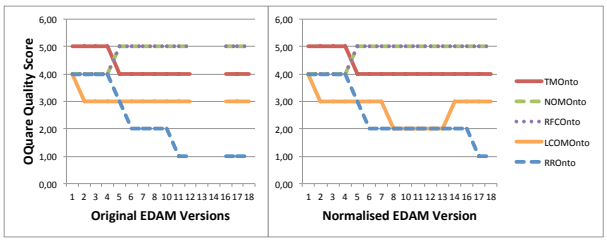


Fig. 2. Graphical representation of those OQuaRE metrics that have changes in scale between the 16 versions of the EDAM ontology

The largest decrease happens for "Formal relation support" (from 4 to 1). This fall is mainly influenced by the behaviour of the RROnto metric, which has 2 changes in scale. The first change is produced by the usage of properties, which descends 86% between v.4 and v.6. The usage of properties also decreases 8% between v.10 and v.11. This variation is smaller than the previous one but, together with an unusual increase in the number of relations (18%), it triggered the change in the RROnto scale. This increase in the number of relations is consequence of a structural change in v.11: deprecated classes were grouped as descendants of an ontology class in the first taxonomic level so the number of relations increased.

It should be noted that RROnto measures the usage of properties, not the number of them. Refactoring towards a common set of properties can often be a good sign, however the usage measures the number of times that a property is linked with an entity through an axiom. For example, while v.4 defines 16 with 6 734 usages, v.5 and v.6 define the same number of properties but with 1 979 and 937 usages respectively.

Finally, the "Structural" characteristic decrease is influenced by the "Tangledness" subcharacteristic. This is associated with TMOnto, which measures the distribution of the parents in the ontology. 10% of the classes have more than 1 direct parent in v.4, while this value grows up to 24% in v.5. This metric has a negative effect over the ontology because of the multiple inheritances, although this might reflect the biology within the ontology.

3.1.3 Influence of deprecated classes: the presence of deprecated classes grows from 3.51% (v.1) to 29.58% (v.18). Deprecated classes caused inconsistencies in v.13 and v.14. Table 3 shows that there are not significant changes at characteristic level between the ontologies with (Org) and without the deprecated classes (Nrm), but some changes happen at metric level. Fig. 2 shows the evolution of the scores for some quality metrics of the complete ontology (left) and the ontology without deprecated classes (right). The structural change previously explained for deprecated classes anticipates the drop of RROnto to v.11, whereas it happens in v.17 in the normalised version. Besides, LCOMOnto temporary descends to score level 2 between v.8 and v.13 in the normalised version. This effect on LCOMOnto cannot be appreciated in the ontologies with the deprecated classes.

3.2 Profiles of activity and its quality scores

Noy *et al.* (2003) state that the study of changes and commonalities should be a complementary process. We interpret the absence of changes as a sign of stability. However, we wonder if this stability is related to the level of activity in the ontology. For example, a difference of 0.12 between two versions might have a different interpretation depending on the number of classes added, edited and removed. We compare OQuaRE quality scores with those obtained with the five potential *profiles of activity* proposed in Malone *et al.* (2010): "initial, ad hoc", "expanding", "refining", "optimising, mature" and "dormant". They provide qualitative descriptions for setting the *profile of activity* of a set of ontologies based on the results obtained with Bubastis, which we applied to our *versioned corpus*. After that, we manually related regions with *profiles of activity* and compare them with our OQuaRE quality scores.

We can observe that some consecutive versions remain unchanged in terms of average quality, and we interpret this as a sign of stability in the quality scores. This happens in the ranges v.2-v.4, v.6-v.10 and v.11-v.18 (see Org columns in Table 3). According to the classification proposed in (Malone and Stevens, 2013), Fig. 3 shows how the ontology starts in a "refining" profile (1-4) because it "is largely refining the classes contained, rather than adding or deleting them, although some addition and deletion still occurs in lower numbers". This stage continues until v.12. From v.4 to

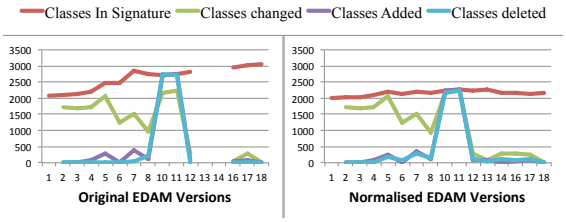


Fig. 3. Number of changes in classes obtained with Bubastis. We have applied Bubastis to every pair of consecutive ontologies θ_{vi} and $\theta_{vi+1} \in L$. We also include the number of classes in signature obtained as a primitive metric with OQuaRE.

v.7 the number of classes increases, so the ontology could be in an "expanding" profile during this stage. However, it seems that these new classes replace deprecated ones (Fig 3 right). Finally, the ontology is in "optimising, mature" stage because there is "no or very low levels of class deletions, some addition of new classes and changes to existing classes". We observe a stability during the "optimising, mature" stage related to the stability in quality scores. At characteristic level, the SD for v.11-v.18 is 0.04 on average, whereas it is 0.17 on average for v.1-v.10.

Finally, this ontology starts in a "refining" stage and evolve to the "optimising, mature". This fact might explain the high quality score and its low variability in EDAM from its first version. Although all the classes in the signature have suffered a change in the URIs between v.8 and v.10, this does not affect the quality scores.

3.2.1 Relation between the ontology status and its quality score: we labeled each version according to the status used by EDAM developers in BioPortal. From v.1 to v.10 they describe EDAM as a beta ontology. A beta version is used to describe a computer artifact that is near completion. The overlap between the stable versions (no beta) and the "optimising, mature" stage is a good indicator.

4 CONCLUSION AND FUTURE WORK

In this work, we have developed a method that combines the analysis of versions with an ontology quality evaluation framework. Its application to EDAM reveals that good ontological principles were applied in its development. The comparison between changes in quality scores and the level of the activity of the ontology justifies the low variability in the scores of the quality characteristics, as EDAM starts in a "refining" stage and evolve to the "optimising, mature" one. The analysis of changes in quality at both subcharacteristic and metric levels have shown some weaknesses and strengths of the ontology and the method. Our approach helps to identify systematically changes based on the OQuaRE metrics. However, it is out of the scope of this work to measure how the changes in quality scores relate to how the content conform to the domain represented by the ontology, which would be the main objective of complementary methods, like realism-based ones (Ceusters and Smith, 2006; Seppälä *et al.*, 2014). As future work, we propose to use the lessons learned in this experiment for improving the sensitivity of the method, in order to be more concise in the detection of changes. We cannot conclude that there is a relation between the quality and activity of classes using one ontology as exemplar, so the analysis of a wider set of ontologies is also a challenge, as it will help us to contextualise OQuaRE scores in the bio-ontology area.

ACKNOWLEDGEMENTS

This project has been possible thanks to the Spanish Ministry of Science and Innovation and the FEDER Programme through grants TIN2010-21388-C02-02, TIN2014-53749-C2-2-R, BES-2011-046192 (MQM) and EEBB-I-14-08700 (MQM), and by the Fundación Séneca (15295/PI/10).

REFERENCES

Ceusters, W. and Smith, B. (2006). A realism-based approach to the evolution of biomedical ontologies. In *AMIA Annual Symposium Proceedings*, volume 2006, page 121. American Medical Informatics Association.

Copeland, M., Gonçalves, R. S., Parsia, B., Sattler, U., and Stevens, R. (2013). Finding fault: detecting issues in a versioned ontology. In *Proceedings of the Second International Workshop on Debugging Ontologies and Ontology Mappings, Montpellier, France, May 27, 2013*, pages 9–20.

Duque-Ramos, A., Fernández-Breis, J. T., Stevens, R., and Aussenac-Gilles, N. (2011). OQuaRE: A square-based approach for evaluating the quality of ontologies. *Journal of Research and Practice in Information Technology*, **43**(2), 159–176.

Gangemi, A., Catenacci, C., Ciaramita, M., and Lehmann, J. (2006). Ontology evaluation and validation. *Proceedings of the 3rd European Semantic Web Conference (ESWC2006)*, **3**, 140–154.

Hoehndorf, R., Haendel, M., Stevens, R., and Rebholz-Schuhmann, D. (2014). Thematic series on biomedical ontologies in JBMS: challenges and new directions. *Journal of biomedical semantics*, **5**, 15.

Klein, M. and Fensel, D. (2001). Ontology versioning on the Semantic Web. *Proc of the Int Semantic Web Working Symposium SWWS*, **31**, 75–91.

Ma, Y., Ma, X., Liu, S., and Jin, B. (2009). A proposal for stable semantic metrics based on evolving ontologies. In *Artificial Intelligence, 2009. JCAI '09. International Joint Conference on*, pages 136–139.

Malone, J. and Stevens, R. (2013). Measuring the level of activity in community built bio-ontologies. *Journal of Biomedical Informatics*, **46**(1), 5–14.

Malone, J., Holloway, E., Adamusiak, T., Kapushesky, M., Zheng, J., Kolesnikov, N., Zhukova, A., Brazma, A., and Parkinson, H. (2010). Modeling sample variables with an Experimental Factor Ontology. *Bioinformatics*, **26**(8), 1112–1118.

Noy, N. F., Musen, M. a., Informatics, S. M., and Drive, C. (2003). Ontology Versioning as an Element of an Ontology-Management Framework. pages 1–17.

Noy, N. F., Kunnatur, S., Klein, M., and Musen, M. A. (2004). Tracking changes during ontology evolution. In S. McIlraith, D. Plexousakis, and F. van Harmelen, editors, *The Semantic Web ISWC 2004*, volume 3298, pages 259–273.

Rogers, J. E. *et al.* (2006). Quality assurance of medical ontologies. *Methods Inf Med*, **45**(3), 267–274.

Seppälä, S., Smith, B., and Ceusters, W. (2014). Applying the realism-based ontology-versioning method for tracking changes in the basic formal ontology. In *Formal Ontology in Information Systems - Proc. of the Eighth International Conference, FOIS 2014, September, 22-25, 2014, R. de Janeiro, Brazil*, pages 227–240.

Stevens, R., Wroe, C., Gobel, C., , and Lord, P. (2008). Applications of ontologies in bioinformatics. In S. Staab and R. Studer, editors, *Handbook on Ontologies in Information Systems*, pages 635–658. Springer.

Tartir, S. and Arpinar, I. B. (2007). Ontology evaluation and ranking using ontoqa. In *ICSC '07: Proceedings of the International Conference on Semantic Computing*, pages 185–192, Washington, DC, USA. IEEE Computer Society.

Tartir, S., Arpinar, I. B., Moore, M., Sheth, A. P., and Aleman-meza, B. (2005). Ontoqa: Metric-based ontology quality analysis. In *IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources*.

Wang, X. H., Zhang, D. Q., Gu, T., and Pung, H. K. (2004). Ontology based context modeling and reasoning using owl. In *Pervasive Computing and Communications Workshops, 2004. Proc. of the 2nd IEEE Annual Conference on*, pages 18–22. Ieee.

Whetzel, P. L., Noy, N., Shah, N., Alexander, P., Dorf, M., Fergerson, R., Storey, M. A., Smith, B., Chute, C., and Musen, M. (2011). BioPortal: Ontologies and integrated data resources at the click of a mouse. *CEUR Workshop Proceedings*, **833**, 292–293.

Yao, H., Orme, A., and Etzkorn, L. (2005). Cohesion metrics for ontology design and application. *Journal of Computer Science*, (1).

Structured Data Acquisition with Ontology-Based Web Forms

Rafael S. Gonçalves*, Samson W. Tu, Csongor I. Nyulas,
Michael J. Tierney and Mark A. Musen

Stanford Center for Biomedical Informatics Research
Stanford University, Stanford, California, USA

ABSTRACT

Structured data acquisition is a common, challenging task that is widely performed in the field of biomedicine. However, in some biomedical fields, such as clinical functional assessment, little effort has been done to structure functional assessment data in such a way that it can be automatically employed in decision making (e.g., determining eligibility for disability benefits) based on conclusions derived from acquired data (e.g., assessment of impaired motor function). In order to be able to apply such automatisms, we need data structured in a way that can be exploited by automated deduction systems, for instance, in the Web Ontology Language (OWL); the *de facto* ontology language for the Web. The rise of OWL caused a paradigm shift in knowledge systems from frame-based to axiom-based. Because of the axiom-based nature of OWL, it is more difficult to acquire instance data based on OWL than it was based on frames. In this paper we tackle the problem of generating Web forms from OWL ontologies, and aggregating input gathered through these forms as an ontology of "semantically-enriched" form data that can be queried using an RDF query language, such as SPARQL. The ontology-based structured data acquisition framework that we have developed is presented through its specific application to the clinical functional assessment domain, with examples of how one can perform desirable analyses of gathered data with simple queries.

1 INTRODUCTION

Ontology-based form generation and structured data acquisition was first pioneered almost 30 years ago. In the early 1990s, Protégé-Frames used definitions of classes in an ontology to generate knowledge-acquisition forms, which could be used to acquire instances of the classes [2, 3]. With OWL as the preferred modeling language for ontologies, class definitions are collections of description logic (DL) axioms, and can no longer be seen as templates for forms [9]. Unlike template-based knowledge representations, where what can be said about a class is defined by the slots of the class template, axiom-based representations do not have this kind of locally scoped specification, and allow any axiom describing the same class to be added to the ontology, as long as the axiom does not lead to inconsistencies. Template-based knowledge representation systems use closed-world reasoning and have local constraints (e.g., cardinality of a slot for a particular class) that can be validated easily, while in an axiom-based system with the open-world assumption such local constraint checking is much more problematic. Furthermore, in our chosen application domain, assessment instruments have specific formats that do not lend themselves to be seen as representing instances of domain ontology classes. Items in the instruments have potentially complex descriptions of information to be collected, such as the severity of pain with a particular quality, and at a specific anatomical location. The challenge is to model the assessment instruments and relate the assessed data to a domain ontology with which one can formulate meaningful queries.

In this paper, we describe a solution for representing, acquiring and querying assessment data that uses (1) domain ontologies and standard terminologies to give formal descriptions of entities in our chosen domain, (2) an information model of assessment instruments to drive the generation of data-acquisition Web forms, and (3) a data model for the acquired information that links the data to the domain ontologies and standard terminologies. Such linkage makes it possible to query and aggregate the data using the logical representation of the domain concepts in the ontologies.

2 RELATED WORK

In addition to the comparison with Protégé-Frames' template-based instance acquisition method described in Section 1, we briefly contrast our work with two other systems that are designed to use forms for acquiring structured data: the first targets the domain of patient assessment, which is similar to the work reported here, while the second is a generic Web-based technology from which one can draw examples on how to arrive at a domain-independent solution.

The clinical documentation system described in [6] uses a template schema to allow a technology-savvy clinician to create documentation templates that include the local structure of subforms and potentially complex clinical descriptions consisting of features and their values. The features and values are mapped to a medical ontology, and the system automatically generates ontological descriptions of the data elements based on the mappings. Constrained by our goal to replicate existing forms, we took the opposite approach where we start with ontological descriptions of the data elements, specify how they are used in assessment instruments as part of the description of instruments, and generate Web forms for the acquisition of data. Having the freedom to design their documentation system, Horridge *et al.* avoided the laborious work of manually modeling the domain concepts.

Semantic wikis extend regular wikis with semantic technologies, wherein each wiki article is an RDF resource, and an instance of some resource such as a class defined in the schema,[1] which can be asserted to have relations with other RDF resources. These relations are defined by the authors of wiki articles, which could be a challenging task to perform without previous knowledge of the domain or the modeling. In a survey of semantic wikis featuring OWL reasoning and SPARQL[2] querying facilities [4], a user

*To whom correspondence should be addressed: rafaelsg@stanford.edu

[1] The typical kinds of schema accepted are OWL and RDFS.

[2] `http://www.w3.org/TR/rdf-sparql-query`

evaluation of a chosen semantic wiki implementation concluded that authoring instance data in such a way is cumbersome, even with users that were familiar with ontologies. A good solution to this would be exploiting the relations defined in the schema to provide "wiki article templates" whose form input fields derive from those relations, thus making it easier to author semantic wiki articles.

3 APPLICATION DOMAIN

Clinical functional assessment provides the application motivation for our work. Functional assessment is the evaluation of an individual's ability to perform body functions (e.g., flexing a joint) and defined tasks (e.g., walking a specific distance). It is necessary for evaluating disabilities for rehabilitation, for social security payment, or for decisions to retain or discharge service members who may be injured on duty. Despite its importance, it is not usually supported by electronic health record (EHR) systems [1]. These assessments are often documented using assessment instruments (e.g., check-lists and validated questionnaires) such as Karnofsky Performance Status [11]. Too frequently the data derived from using these instruments are saved as either blobs or non-standard data elements. While a standard such as LOINC® (Logical Observation Identifiers Names and Codes) defines the syntactic structures of assessment instruments as a hierarchy of panels with questions that have coded answers [10], it does not relate the semantic content of the questions and answers to standard terminologies and data models that allow meaningful querying and aggregation of acquired data.

In our application scenario we use, as exemplars, the U.S. Department of Veterans Affairs (VA) Disability Benefits Questionnaires (DBQs). DBQs are used to evaluate service members' disabilities and to determine the benefits for which they are eligible. We start off with these DBQs as our initial form specifications, and design an ontology-based method for Web form generation and structured data acquisition, subsequently exemplifying how one would go about exploiting such data for immediate or *post facto* analyses.

4 MODELING

In order to capture the semantic distinctions that are needed in functional assessment, we developed a Clinical Functional Assessment (CFA) ontology that models the concepts and relationships that occur in functional assessment instruments. We developed information models for such instruments and for data captured in the instruments. We will show how the CFA ontology and information models inform the generation of data-acquisition forms and how the resulting data can be queried and aggregated. Our goal was to develop a set of light-weight ontologies and models with minimal ontological commitments, and postponing alignment with possible upper-level ontologies to the future. Existing ontologies, such as the Information Artifact Ontology (IAO),[3] do not provide a modeling of forms and questions that we could reuse. Furthermore, what we need is an information model that states, for example, that the structure of a "question" includes a specific text, not an ontology that models parts of information artifacts as ontological entities (e.g., modeling the text of a question as an instance of "textual entity" class). Our ontologies reference the International Classification of Functioning, Disability and Health (ICF),[4] developed by the World Health Organization (WHO), and other reference terminologies such as SNOMED CT.[5]

Imports structure The modeling tasks of this project involve describing different domain areas, leading us to create separate ontology files that can be re-used independently. In our specific application we use the full import closure as depicted in Figure 1.

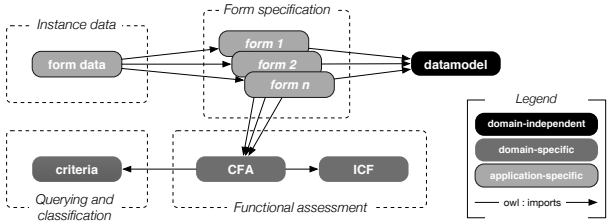


Fig. 1: Imports structure and role separation of ontologies developed for, or included as part of our modeling solution. Form specifications use terms from the *datamodel* ontology (e.g., to create question instances) as well as from domain-specific ontologies (e.g., CFA).

The ontology marked as *Instance data* in Figure 1 is the collection of data assertions from form submissions, possibly from different forms. The ontologies represented in *Form specification* are specifications of different forms; in our case, we use a single ontology that specifies two closely-related forms. The content of the above-mentioned ontologies is application-specific, that is, the way the data is represented is directly derived from the way in which forms are modeled (for different assessment instruments). However, resulting data still conform to the generic information models specified in the *datamodel* ontology. In this way, there is a separation of the *Form specification* ontologies (Abox axioms) from the *Functional assessment* ontologies that model the functional assessment domain and data models (mostly Tbox axioms). In *Querying and classification* we use a domain-specific ontology to apply SWRL rules,[6] and define complex OWL classes to facilitate querying in SPARQL and in OWL.

ICF ICF is a multi-purpose classification that, together with the International Classification of Diseases (ICD),[7] is a reference classification in the WHO Family of International Classifications (WHO–FIC). It provides a standard language and conceptual basis for the definition and measurement of functions and disability. However, unlike ICD codes that represent possible disease or injuries, coding different health and health-related states requires that ICF codes (e.g., "d4501" - walking long distance) be used in conjunction with component-specific qualifiers (e.g., a 0 to 4 scale to encode the range of impairment). Such a complex coding scheme makes it difficult to transform data derived from assessment instruments into the ICF format. Nevertheless, ICF provides a reference conceptual basis for the definition and measurement of functions and disability, thus justifying its usage in descriptions of functional assessment results, despite its limitations

[3] https://code.google.com/p/information-artifact-ontology

[4] http://www.who.int/classifications/icf/en

[5] http://www.ihtsdo.org/snomed-ct

[6] http://www.w3.org/Submission/SWRL

[7] http://www.who.int/classifications/icd/en

as a formal ontology [7]. To reference ICF concepts in our modeling of functional assessment descriptors, we use a version of ICF available from the National Center of Biomedical Ontology (NCBO) BioPortal repository [8], that is represented in OWL.

CFA The Clinical Functional Assessment (CFA) ontology models concepts and relationships that allow us to give formal descriptions of the findings, assessments, and measurements embodied in clinical functional assessment instruments. The ontology is divided into three main branches: (1) *Finding*: the result of an observation or judgement, (2) *Value* that defines collections of possible qualifiers and values for findings, and (3) *SubjectMatterOntology* that provides internally defined domain concepts that either are not available from standard terminologies or are references to standard terms that need to be organized into taxonomies. The *Finding* class is further subdivided into *Assessment* (those findings that have non-numeric result) and *Measurement* (those findings that have numeric results). We also define *FunctionalFinding* (a subclass of *Finding*) and *FunctionalAssessment* (a subclass of *Assessment*). In general, a functional assessment will have some assessed function that can be related to an ICF body function or activity (possibly as an exact match, specialization, or generalization), some assessed attribute, such as severity, that specifies the dimension of the function being assessed, and optionally some anatomical location of the assessment. Both findings and functions can be modified by qualifiers that further refine these entities. For example, a functional assessment may be made in the context of using assistive devices, and a function being assessed may have some temporal component (e.g., constant or intermittent pain). ICF being an imported ontology for CFA, all ICF categories, such as body structure, body function, activities and participation, and environmental factors are available for formalizing descriptions of functional assessments. For other standard terminologies such as SNOMED CT, ICD, and LOINC, instead of importing them as ontologies, we make references to them through an *ExternallyCodedValue* that specifies the terminology source and code. Queries that reference these codes require the availability of terminology services that relate these codes to other terms in the referenced terminologies.

The modeling of *Finding* is exemplified as follows, based on the "Back (Thoracolumbar Spine) Conditions" DBQ that we use as one of our exemplar assessment instruments; in the question on the severity of constant pain caused by radiculopathy on the right lower extremity, we define a subclass of *FunctionalAssessment* that has the assessed attribute 'severity', the assessed function 'icf:b2801 Pain in body part' that is qualified by a temporal quality 'Constant', and has anatomical location 'icf:s750. structure of lower extremity' with laterality 'Right'. Figure 2 illustrates the modeling of this assessment. With the modeling of the dimensions of assessment instrument questions, we can make queries on, and aggregate data collected through the instruments, as will be shown in Section 6.

```
cfa:hasAnatomicalLocation some
    ('icf:s750. Structure of lower extremity'
     and (cfa:hasLaterality value cfa:Right))
cfa:hasAssessedFunction some
    (cfa:Function
     and (cfa:isExactMatchOf some 'icf:b2801. Pain in body part')
     and (cfa:causedBy value cfa:Radiculopathy)
     and (cfa:hasQualifier value cfa:Constant))
cfa:SeverityOfPainSensationCausedByRadiculopathy
```

Fig. 2: Modeling of "severity of constant pain caused by radiculopathy in the lower right extremity".

Datamodel The *datamodel* ontology is a generic, context-free representation of a form (e.g., it models elements such as questions and sections) and the data generated from a form (e.g., a string value from a text area, or values from an enumerated value set). Figure 3 summarizes key aspects of our modeling: elements of a form are asserted as subclasses of *FormStructure*, such as *Form*, *Section* and *Question*. Each kind of *FormStructure* generates some kind of *Data*; every form submission generates an instance of *FormData*, which references (via the *hasComponent* property) all instances of *Data* generated in the process of parsing form answers. Specific sections such as *SubjectInfoSection* collect information pertaining to a subject, and these details are aggregated in an instance of *SubjectInformation*. An answer to an instance of *Question* gives rise to an instance of *Observation* with a *hasValue* property assertion to the IRI of the selected answer. An instance of *Observation* will be inferred to have an outgoing *hasFocus* property assertion if the *Question* instance it derives from encodes some kind of semantic description of the question's meaning via the *isAbout* relation. Each instance of *Question* specifies a set of possible (answer) values via a *hasPossibleValue* relation to a subclass of *Value*.

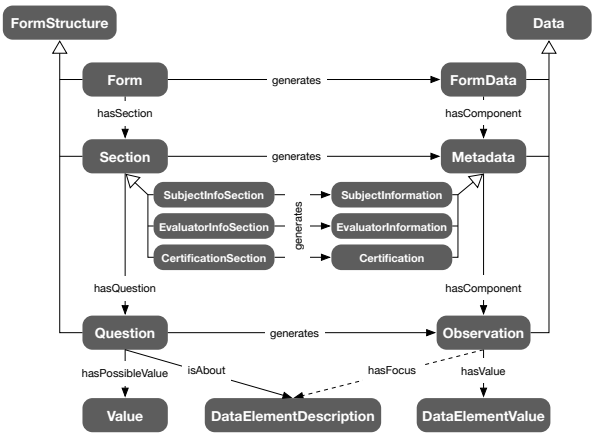


Fig. 3: Excerpt of the *datamodel* ontology classes and relations.

Form The *Form* ontology contains the set of individuals that are necessary to produce forms. While the technology we have developed is completely generic, we use as exemplars the U.S. Department of Veterans Affairs (VA) DBQs, which we modeled in an ontology named *DBQ*. This ontology contains instances of *Question*, *Section*, *Form* and other elements defined in the *datamodel* ontology (shown in Figure 3). Not only does this ontology rely on *datamodel* (for form structuring purposes), it also relies on functional assessment classes and individuals given in the *CFA* ontology, for example, values of a scale of severity of pain that should be presented as answer options to users reporting on the severity of constant pain in the lower extremity.

Criteria The *criteria* ontology contains SWRL rules to enrich the domain representation (e.g., if a *Question* instance has an *isAbout* relation with some instance i, then the *Observation* data instance that represents the answer to that question will get a *hasFocus* property filler i), as well as defined classes used to better support querying, which we describe in more detail in Section 6.

5 OWL-BASED DATA ACQUISITION

Our approach to data acquisition in OWL requires two components: firstly, an OWL representation (in the form of one or more ontologies) of the form structures (questions, sections, etc), and descriptions of those structures' meanings, and, secondly, the view component that is given by an XML file specifying user-interface aspects. So, in order to use our method, a user will have to model questions and their descriptions in OWL, and then specify the layout and content of the resulting form in XML.

We implemented our form generation and data acquisition tool in Java, using the OWL API v4.0.1,[8] and its source code is publicly available on GitHub.[9] The tool implementation and configuration details are omitted here due to lack of space, but can be found in the GitHub project wiki. The tool takes as input a user-defined XML configuration file, generates a form, and outputs form answers in CSV, RDF and OWL formats. The configuration file should contain a pointer to the ontology specifying the form, as well as its imports. The two major stages in the service are form generation and form input handling, as described below.

(1) Form generation – Steps to produce a form:
  (a) Process XML configuration, gathering form layout information, IRIs and bindings to ontology entities
  (b) Extract from the input ontology all relevant information pertaining to each form element:
    (b.1) Text to be displayed (e.g., section header, question text)
    (b.2) Options and their text, where applicable
    (b.3) The focus of each question
  (c) Generate the appropriate HTML and JavaScript code
(2) Form input handling – Once the form is filled in and submitted:
  (a) Process answer data and create appropriate individuals
  (b) Produce a partonomy of the individuals created in (2.a) that mirrors the layout structure given in the configuration
  (c) Return the (structured) answers to the user in a chosen format

The user-defined XML configuration (1.a) specifies: input and output information of the tool, bindings to ontology entities, and layout of form elements. The key XML elements are:

input: contains an *ontology* child element, and optionally a child element named *imports*
- ***ontology***: absolute path or URL to the form specification ontology (e.g., *DBQ ontology*)
- ***imports***: contains *ontology* child elements, which have an attribute *iri*, giving the IRI of the imported ontology

output: contains the following child elements
- ***file***: defines, via a *title* attribute, the title of the form. Optionally, a path can be specified within the *file* element where the HTML form file should be serialized
- ***cssStyle***: the CSS style class to be used in the output HTML

bindings: defines mappings to ontology entities, such as what data property is used to state the text of a question, or section headings

form: defines the layout and behaviors of the form

There is a wide range of versatility when configuring forms, such as: multiple levels of sub-questions, form element numbering, question type (e.g., radio, checkbox, dropdown, horizontal checkbox, etc), question-list layout (vertical or inline) and recurrence; one can specify that a collection of questions should be repeated any given number of times. Some more complex options include overriding the default (alphabetic) order of answer options, and triggering sub-questions when a specific answer is selected. These two features are exemplified in Figure 4: this question is configured with an attribute/value pair: *showSubquestionsForAnswer*="cfa:Yes" on the *question* XML element, so that answering 'Yes' triggers the sub-questions of that question. In Figure 4, under 'Right lower extremity', we have a question with a list of answer options derived from an enumerated value set, which would ordinarily be ordered alphabetically. However, 'None' would then appear between 'Moderate' and 'Severe', thus interrupting a severity scale. So we added: *optionOrder*="3;*" to the *question* element, which states that the would-be third option (alphabetically) should appear first, and the remaining (the "*" wild character stands for "all unmentioned options") should be presented in default order.

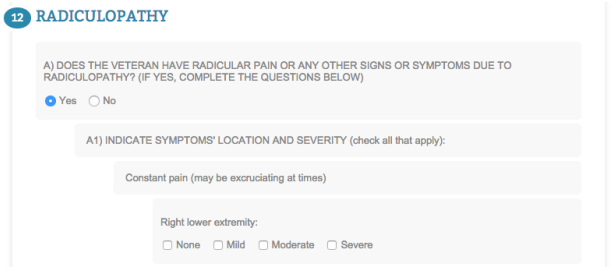


Fig. 4: The user interface of the form generated for the DBQ question corresponding to radiculopathy pain modeled in Figure 2.

The key output of the data acquisition tool is the OWL ontology, as it provides us with "semantically enriched" form data that can be used for aggregation and querying. The resulting data individuals are structured in OWL (via *hasComponent* relations) similarly to how the form is structured in the configuration, that is, if question Q is configured as having two sub-questions, then the *Observation* individual generated by Q will have two outgoing *hasComponent* relations to the instances of *Observation* generated by the two sub-questions of Q.

6 DATA ANALYSIS

One of the authors (Michael J. Tierney), who is a physician from the VA Palo Alto Healthcare System, validated the generated OWL-based versions of the DBQ forms, and filled in the "Back (Thoracolumbar Spine) Conditions" DBQ with 5 complete sets of sample data. The data gathered are stored in a graph database with support for SPARQL 1.1 querying and OWL 2 reasoning.

Since our data are both structured and semantically enriched, we are able to query the observations using SPARQL, classify them into criteria representing powerful OWL expressions, or manipulate them using SWRL. For example, Code Snippet 1 presents a simple SPARQL query that returns all instances of *Observation* where a patient presented signs or symptoms due to radiculopathy. It is worth observing that this query is formulated in such a way that it is independent of the assessment instrument, including the particular formulation of the question, but rather uses the appropriate focus individual from our CFA ontology.

[8] http://owlapi.sourceforge.net

[9] http://github.com/protegeproject/facsimile

Code Snippet 1 SPARQL query for retrieving all observations of radicular pain due to radiculopathy.

```
SELECT ?obs WHERE {
  ?obs a datamodel:Observation .
  ?obs datamodel:isDerivedFrom ?q .
  ?q a datamodel:Question .
  ?q cfa:isAbout
     cfa:signs_or_symptoms_due_to_radiculopathy .
  ?obs cfa:hasValue cfa:Yes }
```

In order to query for all observations of severe pain anywhere in the lower extremity, one could formulate an OWL DL query such as that given in Code Snippet 2.

Code Snippet 2 OWL DL query for retrieving all observations of severe pain anywhere in the lower extremity.

```
datamodel:Observation and
cfa:hasValue value cfa:severe and
cfa:hasFocus some (cfa:Assessment and
    (cfa:hasAssessedFunction some
        (cfa:isExactMatchOf some
            'icf:b2801. Pain in body part')) and
    (cfa:hasAnatomicalLocation some
        'icf:s750. Structure of lower extremity'))
```

In response to the query in Code Snippet 2, a DL reasoner uses the semantic descriptions of the observation foci, which are derived from the questions' *isAbout* property, to aggregate answers for severe pain for different parts of the lower extremity.

7 DISCUSSION

In this paper we presented a framework for OWL-based form generation and data acquisition that gathers form answers as tab-delimited data, RDF triples, or OWL instances, which can be subsequently analyzed in a systematic way (as shown in our queries in Section 6). Once the raw data is processed (by deriving the foci of observations from the *isAbout* field of the questions), the resulting data have no dependency on specific questions (except for provenance tracking), so if the form specification is modified, then previous form data are still comprehensible and sound (i.e., upon form specification changes the new data and old data remain compatible). However, if a user requires data to be structured in a different or more specialized format than ours, then either the software needs modifying, or a post-processing step would be necessary. The value of data in such a structured format in any arbitrary domain is twofold: automating, or improving the automation of the process of arriving at desirable conclusions from questions in the form, and for further analysis, for instance, via querying. In the clinical functional assessment domain, our modeling of forms and questions is consistent with the format of assessment instruments defined in LOINC. However, the types of queries we formulated for functional assessment data are unfeasible using LOINC, since LOINC provides no semantics behind what an answer to a specific question means.

We presented our modeling of functional assessments and assessment instruments, and demonstrated (1) how to generate forms and acquire data based on these OWL ontologies and data models, and (2) how to make use of the data using queries on individual subjects and queries that aggregate population data. The modeling contributions include (1) *CFA*: a clinical functional assessment domain ontology that allows defining questions being asked in an assessment instrument in terms of a rich ontology that integrates standard terminologies such as ICF and SNOMED CT, and which provides the means for making detailed or aggregate queries on acquired data, and (2) *datamodel*: an information model that allows the specification of generic assessment forms and the format of structured data acquired through the instruments.

We have designed our output model to support the acquisition of structured data through Web forms, and for the potential to integrate the data inside EHRs. It is straightforward to transform the data we capture as instances of *Observation*, *Certification*, *EvaluatorInformation*, and *SubjectInformation* into, for example, Health Level Seven (HL7) Reference Information Model (RIM) standard compliant data [5]. Finally, we have shown that the problem of structured data acquisition can be suitably tackled using OWL; our solution, though applied to the clinical functional assessment domain for the context of this paper, is entirely generic, and can easily be applied to an arbitrary domain.

ACKNOWLEDGMENTS

This work is supported in part by contract W81XWH-13-2-0010 from the U.S. Department of Defense, and grants GM086587 and GM103316 from the U.S. National Institutes of Health (NIH).

REFERENCES

[1] Buyl, R. and Nyssen, M. (2009). Structured electronic physiotherapy records. *Int. J. of Med. Inf.*, **78**(7), 473–481.

[2] Eriksson, H., Puerta, A. R., and Musen, M. A. (1994). Generation of knowledge-acquisition tools from domain ontologies. *Int. J. of Human-Computer Studies*, **41**, 425–453.

[3] Gennari, J. H., Musen, M. A., Fergerson, R. W., Grosso, W. E., Crubzy, M., *et al.* (2003). The evolution of Protégé: an environment for knowledge-based systems development. *Int. J. of Human-Computer Studies*, **58**(1), 89–123.

[4] Gonçalves, R. S. (2009). *Semantic Wiki for Travel and Holidays using OWL*. Master's thesis, The University of Manchester.

[5] Health Level Seven (2015). HL7 Reference Information Model. `www.hl7.org/implement/standards/rim.cfm`.

[6] Horridge, M., Brandt, S., Parsia, B., and Rector, A. (2014). A domain specific ontology authoring environment for a clinical documentation system. In *Proc. of CBMS-14*.

[7] Kumar, A. and Smith, B. (2005). The ontology of processes and functions: A study of the international classification of functioning, disability and health. In *Proc. of the AIME Workshop on Biomedical Ontology Engineering*.

[8] Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., *et al.* (2009). BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, **37**, 170–173.

[9] Rector, A. (2013). Axioms & templates: Distinctions & transformations amongst ontologies, frames & information models. In *Proc. of K-CAP-13*.

[10] Vreeman, D. J., McDonald, C. J., and Huff, S. M. (2010). Representing patient assessments in LOINC®. In *Proc. of AMIA*.

[11] Yates, J. W., Chalmer, B., McKegney, F. P., *et al.* (1980). Evaluation of patients with advanced cancer using the Karnofsky performance status. *CANCER*, **45**(8), 2220–2224.

Using Aber-OWL for fast and scalable reasoning over BioPortal ontologies

Luke Slater [1]*, Georgios V Gkoutos[2], Paul N Schofield[3], Robert Hoehndorf[1]

[1] Computational Bioscience Research Center, King Abdullah University of Science and Technology, 4700 KAUST, 23955-6900, Thuwal, Saudi Arabia

[2] Department of Computer Science, Aberystwyth University, Aberystwyth, SY23 3DB, Wales, United Kingdom

[3] Department of Physiology, Development and Neuroscience, University of Cambridge, Downing Street, CB2 3EG, England, United Kingdom

ABSTRACT

Reasoning over biomedical ontologies using their OWL semantics has traditionally been a challenging task due to the high theoretical complexity of OWL-based automated reasoning. As a consequence, ontology repositories, as well as most other tools utilizing ontologies, either provide access to ontologies without use of automated reasoning, or limit the number of ontologies for which automated reasoning-based access is provided. We apply the Aber-OWL infrastructure to provide automated reasoning-based access to all accessible and consistent ontologies in BioPortal (368 ontologies). We perform an extensive performance evaluation to determine query times, both for queries of different complexity as well as for queries that are performed in parallel over the ontologies. We demonstrate that, with the exception of a few ontologies, even complex and parallel queries can now be answered in milliseconds, therefore allowing automated reasoning to be used on a large scale, to run in parallel, and with rapid response times.

1 INTRODUCTION

Major ontology repositories such as the BioPortal (Noy *et al.*, 2009), OntoBee (Xiang *et al.*, 2011), or the Ontology Lookup Service (Cote *et al.*, 2006), have existed for a number of years, and currently contain several hundred ontologies, enabling ontology creators and maintainers to publish their ontology releases and make them available to the wider community.

Besides the hosting functionality that such repositories offer, they usually also provide certain web-based features for browsing, comparing, visualising and processing ontologies. One particularly useful feature, currently missing from the major ontology repositories, is the ability to provide online access to reasoning services simultaneously over many ontologies. Such a feature would enable the use of semantics and deductive inference when processing data characterized with the ontologies these repositories contain (Hoehndorf *et al.*, 2015). Moreover, the ability to query multiple ontologies simultaneously further enables data integration across domains and data sources. For example, there is an increasing amount of RDF (Manola and Miller, 2004) data becoming available through public SPARQL (Seaborne and Prud'hommeaux, 2008) endpoints (Jupp *et al.*, 2014; The Uniprot Consortium, 2007; Belleau *et al.*, 2008; Williams *et al.*, 2012), which utilise multiple ontologies to annotate entities.

*To whom correspondence should be addressed: luke.slater@kaust.edu.sa

However, enabling automated reasoning over multiple ontologies is a challenging task since as automated reasoning can be highly complex and costly in terms of time and memory consumption (Tobies, 2000). In particular, ontologies formulated in the Web Ontology Language (OWL) (Grau *et al.*, 2008) can utilize statements based on highly expressive description logics (Horrocks *et al.*, 2000), and therefore queries that utilize automated reasoning cannot, in general, be guaranteed to finish in a reasonable amount of time.

Prior work on large-scale automated reasoning over biomedical ontologies has often focused on the set of ontologies in Bioportal, as it is one of the largest collections of ontologies freely available. To enable inferences over this set of ontologies, modularization techniques have been applied (Del Vescovo *et al.*, 2011) using the notion of locality-based modules, and demonstrated that, for most ontologies and applications, relatively small modules can be extracted over which queries can be answered more efficiently. Other work has focused on predicting the performance of reasoners when applied to the set of BioPortal ontologies (Sazonau *et al.*, 2013), and could demonstrate that performance of particular reasoners can reliably be predicted; at the same time, the authors have conducted an extensive evaluation of average *classification* times of each ontology.

Other approaches apply RDFS reasoning (Patel-Schneider *et al.*, 2004) for providing limited, yet fast, inference capabilities in answering queries over Bioportal's set of ontologies through a SPARQL interface (Salvadores *et al.*, 2012, 2013). Alternatively, systems such as OntoQuery (Tudose *et al.*, 2013) provide access to ontologies through automated reasoning but limit the number of ontologies.

The Aber-OWL (Hoehndorf *et al.*, 2015) system is a novel ontology repository that aims to allow access to multiple ontologies through automated reasoning utilizing the OWL semantics of the ontologies. Aber-OWL mitigates the complexity challenge by using a reasoner which supports only a subset of OWL (i.e., the OWL EL profile (Motik *et al.*, 2009)), ignoring ontology axioms and queries that do not fall within this subset. This enables the provision of polynomial-time reasoning, which is sufficiently fast for many practical uses even when applied to large ontologies. However, thus far, the Aber-OWL software is only applied to a few, manually selected, ontologies, and therefore does not have a similar coverage as other ontology repositories, nor does it cater for reasoning over large sets of ontologies such as the ones provided by the BioPortal ontology dataset (Bioportal contains, as of 9 March 2015, 428 ontologies consisting of 6,668,991 classes).

Here, we apply the Aber-OWL framework to reason over the majority of the available ontologies in Bioportal. We evaluate the performance of querying ontologies with Aber-OWL, utilizing 337 ontologies from BioPortal, we evaluate Aber-OWL's ability to perform different types of queries as well as its scalability in performing queries that are executed in parallel. We demonstrate that the Aber-OWL framework makes it possible to provide, at least, light-weight description logic reasoning over most of the freely accessible ontologies contained in BioPortal, with a relatively low memory footprint and high scalability in respect to the number of queries executed in parallel, using only a single medium-sized server as hardware to provide these services. Furthermore, we identify several ontologies for which querying using automated reasoning performs significantly worse than the majority of the other ontologies tested, and discuss potential explanations and solutions.

2 METHODS

2.1 Selection of ontologies

We selected all ontologies contained in BioPortal as candidate ontologies, and attempted to download the current versions of all the ontologies for which a download link was provided by BioPortal. A summary of the results is presented in Table 1.

Total	427
Loadable	368
Used	337
Unobtainable	39
Non-parseable	17
Inconsistent	3
No Labels	31

Table 1. Summary of Ontologies used in our test. The loadable ontologies are the ones obtained from BioPortal which could be parsed using the OWL API and which were found to be consistent when classified with the ELK reasoner. We exclude 31 ontologies that do not contain any labels from our analysis.

Out of 427 total ontologies listed by Bioportal, only 368 could be directly downloaded and processed by Aber-OWL. Reasons for failure to load ontologies include the absence of a download link for listed ontologies, proprietary access to ontologies or ontologies that are only available in proprietary data formats (e.g., some of the ontologies and vocabularies provided as part of the Unified Medical Language Systems (Bodenreider, 2004)). 39 ontologies were not obtainable. Furthermore, 17 ontologies that could be downloaded were not parseable with the OWL API, indicating a problem in the file format used to distribute the ontology. Three ontologies were inconsistent at the reasoning stage. Several ontologies also referred to unobtainable ontologies as imports; however, we included these ontologies in our analysis, utilizing only the classes and axioms that were accessible. As Aber-OWL currently relies on the use of labels to construct queries, we further removed 31 ontologies that did not include any labels from our test set.

Overall, we use set of 337 ontologies in our experiments consisting of 3,466,912 classes and 6,997,872 logical axioms (of which 12,721 are axioms involving relations, i.e., RBox axioms). In comparison, BioPortal currently (9 March 2015) includes a total of 6,668,991 classes.

2.2 Use of the Aber-OWL reasoning infrastructure

Aber-OWL (Hoehndorf *et al.*, 2015) is an ontology repository and query service built on the OWLAPI (Horridge *et al.*, 2007) library, which allows access to a number of ontologies through automated reasoning. In particular, Aber-OWL allows users or software applications to query the loaded ontologies using Manchester OWL Syntax (Horridge *et al.*, 2006), using the class and property labels as short-form identifiers for classes. Aber-OWL exposes this functionality on the Internet through a JSON API as well as a web interface available on `http://aber-owl.net`. To answer queries, Aber-OWL utilizes the ELK reasoner (Kazakov *et al.*, 2014, 2011), a highly optimized reasoner that supports the OWL-EL profile. Ontologies which are not OWL-EL are automatically transmuted by the reasoner by means of ignoring all non-EL axioms, though as of 2013 50.7% of ontologies in Bioportal were natively using it (Matentzoglu *et al.*, 2013).

We extended the Aber-OWL framework to obtain a list of ontologies from the Bioportal repository, periodically checking for new ontologies as well as for new versions of existing ontologies. As a result, our testing version of Aber-OWL maintains a mirror of the accessible ontologies available in BioPortal. Furthermore, similarly to the functionality provided by BioPortal, a record of older versions of ontologies is kept within Aber-OWL, so that, in the future, the semantic difference between ontology versions could be computed.

In addition, we expanded the Aber-OWL software to count and provide statistics about:

- The ontologies which failed to load, with associated error messages;
- Axioms, axiom types, and number of classes per ontology; and
- Axioms, axiom types, and number of classes over all ontologies contained within Aber-OWL.

For each query to Aber-OWL, we also provide the query execution time within Aber-OWL and pass this information back to the client along with the result-set of the query.

All information is available through Aber-OWL's JSON API, and the source code freely available at `https://github.com/bio-ontology-research-group/AberOWL`.

2.3 Experimental setup

In order to evaluate the performance of querying single and multiple ontologies in Aber-OWL, randomly queries of different complexity were generated and executed. Since the ELK reasoner utilises a cache for answering queries that have already been computed, each of the generated query consisted of a new class expression. The following types of class expressions were used in the generated queries (for randomly generated `A`, `B`, and `R`):

- Primitive class: `A`
- Conjunctive query: `A and B`
- Existential query: `R some A`
- Conjunctive existential query: `A and R some B`

300 random queries for each of these type were generated for each ontology that was tested (1,200 queries in total per ontology). Each set of the 300 random queries that was generated, was subsequently

split into three sets each of which contained 100 class expressions. The random class expressions contained in the resulting sets were then utilised to perform superclass (100 queries), equivalent (100 queries) and subclass (100 queries) queries and the response time of the Aber-OWL framework was recorded for each of the query.

We further test the scalability of answering the queries by performing these queries in parallel. For this purpose, we remotely query Aber-OWL with one query at once, 100 queries in parallel, and 1,000 queries in parllel.

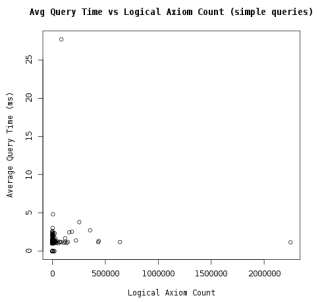
In our test, we record the response time of each query, based on the statistics provided by the Aber-OWL server; in particular, response time does not include network latency. All tests are performed on a server with 128GB memory and two Intel Xeon E5-2680v2 10-core 2.8GHz CPUs with hyper-threading activated (resulting in 40 virtual cores). The ELK reasoner underlying Aber-OWL is permitted to use all available (i.e., all 40) cores to perform classification and respond to queries.

3 RESULTS AND DISCUSSION

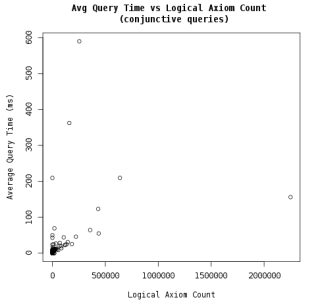
On average, when performing a single query over Aber-OWL, query results are returned in 10.8 milliseconds (standard deviation: 48.0 milliseconds). The time required to answer a query using Aber-OWL correlates linearly with the number of logical axioms in the ontologies (Pearson correlation, $\rho = 0.33$), and also strongly correlates with the number of queries performed in parallel (Pearson correlation, $\rho = 0.82$). Figure 1 shows the query times for the ontologies based on the type of query, and Figure 2 shows the query times based on different number of queries run in parallel. The maximum observed memory consumption for the Aber-OWL server while performing these tests was 66.1 GB.

We observe several ontologies for which query times are significantly higher than for the other ontologies. The most prevalent outlier is the NCI Thesaurus (Sioutos *et al.*, 2007) for which average query time is 600 ms when performing a single query over Aber-OWL. Previous analysis of NCI Thesaurus has identified axioms which heavily impact the performance of classification for the ontology using multiple description logic reasoners (Gonçalves *et al.*, 2011). The same analysis has also shown that it can significantly improve reasoning time to add inferred axioms to the ontology. To test whether this would also allow us to improve reasoning time over the NCI Thesaurus in Aber-OWL and using the ELK reasoner, we apply the Elvira modularization software (Hoehndorf *et al.*, 2011), using the HermiT reasoner to classify the NCI Thesaurus and adding all inferred axioms that fall into the OWL-EL profile to the ontology, as opposed to ELK's approach of ignoring non-EL axioms during classification. We then repeat our experiments. Figure 3 shows the different reasoning times for NCI Thesaurus before and after processing with Elvira. Query time reduces from 703 ms (standard deviation: 689 ms) before processing with Elvira to 51 ms (standard deviation: 42 ms) after processing with Elvira, demonstrating that adding inferred axioms and removing axioms that do not fall in the OWL-EL profile can be used to improve query time.

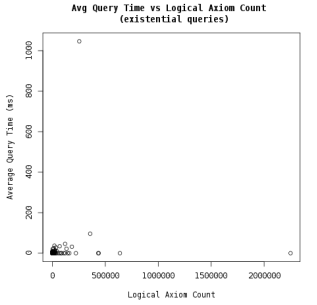
Another outlier with regard to average query time is the Natural Products Ontology (NATPRO, `http://bioportal.bioontology.org/ontologies/NATPRO`). However, as NATPRO is expressed in OWL-Full, it cannot reliably be classified with a Description Logic reasoner, and therefore we cannot apply



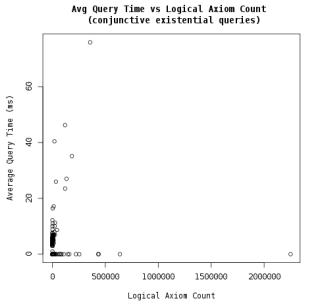
(a) primitive classes



(b) conjunctive queries

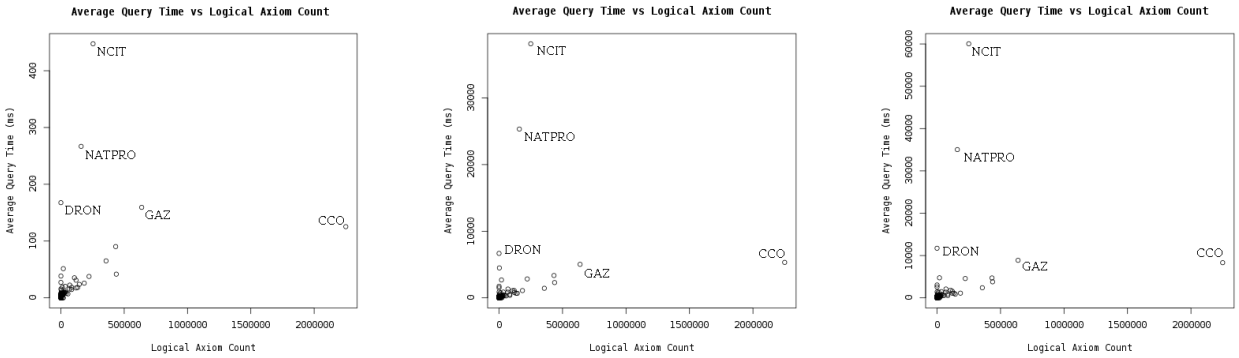


(c) existential queries



(d) conjunctive existential queries

Fig. 1: Query times as function of the number of logical axioms in the ontologies, separated by the type of query.



(a) Sequential querying (b) 100 parallel queries (c) 1,000 parallel queries

Fig. 2: Query times as function of the number of logical axioms in the ontologies, separated by the number of queries executed in parallel.

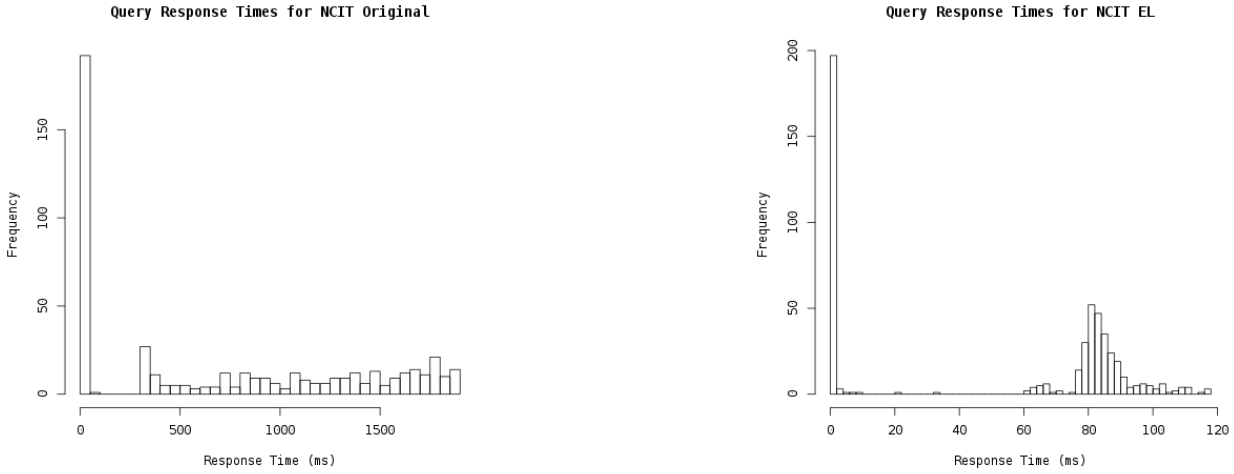


Fig. 3: Query times over the NCI Thesaurus.

the same approach to improve the performance of responding to queries.

3.1 Future Work

The performance of using automated reasoning for querying ontologies relies heavily on the type of reasoner used. We have used the ELK (Kazakov *et al.*, 2014, 2011) reasoner in our evaluation; however, it is possible to substitute ELK with any other OWLAPI-compatible reasoners. In particular, novel reasoners such as Konklude (Steigmiller *et al.*, 2014), which outperform ELK in many tasks (Bail *et al.*, 2014), may provide further improvements in performance and scalability.

We identified several ontologies as leading to performance problems, i.e., they are outliers during query time testing. For these ontologies, including the Natural Products Ontology (NATPRO), and, to a lesser degree, the Drug Ontology (DRON) (Hanna *et al.*, 2013), similar 'culprit-finding' analysis methods may be applied as have previously been applied for the NCI Thesaurus (Gonçalves *et al.*, 2011). These methods may also allow the ontology maintainers to identifying possible modifications to their ontologies that would result in better reasoner performance.

4 CONCLUSION

We have demonstrated that it is feasible to reason over most of the ontologies available in BioPortal in real time, and that queries over these ontologies can be answered quickly, in real-time, and using only standard server hardware. We further tested the performance of answering queries in parallel, and show that, for the majority of cases, even highly parallel access allows quick response times.

We have also identified a number of ontologies for which performance of automated reasoning, at least when using Aber-OWL and the ELK reasoner, is significantly worse, which renders them particularly problematic for application that carry heavy parallel loads. At least for some of these ontologies, pre-processing ontologies using tools such as Elvira (Hoehndorf *et al.*, 2011) can mitigate these problems.

The ability to reason over a very large number of ontologies, such as all the ontologies in BioPortal, opens up the possibility to frequently use reasoning not only locally when making changes to a single ontology, but also monitor – in real time – the consequences that a change may have on other ontologies, in particular on ontologies that may import the ontologies that is being changed. Using automated reasoning over all ontologies within a domain therefore has the potential to increase interoperability between ontologies and associated data by verifying mutual consistency and enabling queries across multiple ontologies, and our results show that such a system can now be implemented with the available software tools and commonly used server hardware.

ACKNOWLEDGEMENTS

REFERENCES

Bail, S., Glimm, B., Jiménez-Ruiz, E., Matentzoglu, N., Parsia, B., and Steigmiller, A., editors (2014). *ORE 2014: OWL Reasoner Evaluation Workshop*. Number 1207 in CEUR Workshop Proceedings. CEUR-WS.org, Aachen, Germany.

Belleau, F., Nolin, M., Tourigny, N., Rigault, P., and Morissette, J. (2008). Bio2RDF: Towards a mashup to build bioinformatics knowledge systems. *Journal of Biomedical Informatics*, **41**(5), 706–716.

Bodenreider, O. (2004). The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res*, **32**(Database issue), D267–D270.

Cote, R., Jones, P., Apweiler, R., and Hermjakob, H. (2006). The ontology lookup service, a lightweight cross-platform tool for controlled vocabulary queries. *BMC Bioinformatics*, **7**(1), 97+.

Del Vescovo, C., Gessler, D. D., Klinov, P., Parsia, B., Sattler, U., Schneider, T., and Winget, A. (2011). Decomposition and modular structure of bioportal ontologies. In *The Semantic Web–ISWC 2011*, pages 130–145. Springer.

Gonçalves, R. S., Parsia, B., and Sattler, U. (2011). Analysing multiple versions of an ontology: A study of the nci thesaurus. In *24th International Workshop on Description Logics*, page 147. Citeseer.

Grau, B., Horrocks, I., Motik, B., Parsia, B., Patelschneider, P., and Sattler, U. (2008). OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web*, **6**(4), 309–322.

Hanna, J., Joseph, E., Brochhausen, M., and Hogan, W. (2013). Building a drug ontology based on rxnorm and other sources. *Journal of Biomedical Semantics*, **4**(1), 44.

Hoehndorf, R., Dumontier, M., Oellrich, A., Wimalaratne, S., Rebholz-Schuhmann, D., Schofield, P., and Gkoutos, G. V. (2011). A common layer of interoperability for biomedical ontologies based on OWL EL. *Bioinformatics*, **27**(7), 1001–1008.

Hoehndorf, R., Slater, L., Schofield, P. N., and Gkoutos, G. V. (2015). Aber-owl: a framework for ontology-based data access in biology. *BMC Bioinformatics*.

Horridge, M., Drummond, N., Goodwin, J., Rector, A., Stevens, R., and Wang, H. (2006). The Manchester OWL Syntax. *Proc. of the 2006 OWL Experiences and Directions Workshop (OWL-ED2006)*.

Horridge, M., Bechhofer, S., and Noppens, O. (2007). Igniting the OWL 1.1 touch paper: The OWL API. In *Proceedings of OWLED 2007: Third International Workshop on OWL Experiences and Directions*.

Horrocks, I., Sattler, U., and Tobies, S. (2000). Practical reasoning for very expressive description logics. *Logic Journal of the IGPL*, **8**(3), 239–264.

Jupp, S., Malone, J., Bolleman, J., Brandizi, M., Davies, M., Garcia, L., Gaulton, A., Gehant, S., Laibe, C., Redaschi, N., Wimalaratne, S. M., Martin, M., Le Novre, N., Parkinson, H., Birney, E., and Jenkinson, A. M. (2014). The EBI RDF platform: linked open data for the life sciences. *Bioinformatics*, **30**(9), 1338–1339.

Kazakov, Y., Krötzsch, M., and Simančík, F. (2011). Unchain my $\mathcal{EL}$ reasoner. In *Proceedings of the 23rd International Workshop on Description Logics (DL'10)*, CEUR Workshop Proceedings. CEUR-WS.org.

Kazakov, Y., Krötzsch, M., and Simancik, F. (2014). The incredible elk. *Journal of Automated Reasoning*, **53**(1), 1–61.

Manola, F. and Miller, E., editors (2004). *RDF Primer*. W3C Recommendation. World Wide Web Consortium.

Matentzoglu, N., Bail, S., and Parsia, B. (2013). A corpus of owl dl ontologies. *Description Logics*, **1014**, 829–841.

Motik, B., Grau, B. C., Horrocks, I., Wu, Z., Fokoue, A., and Lutz, C. (2009). Owl 2 web ontology language: Profiles. Recommendation, World Wide Web Consortium (W3C).

Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A. A., Chute, C. G., and Musen, M. A. (2009). Bioportal: ontologies and integrated data resources at the click of a mouse. *Nucleic acids research*, **37**(Web Server issue), W170–173.

Patel-Schneider, P. F., Hayes, P., and Horrocks, I. (2004). Owl web ontology language semantics and abstract syntax section 5. rdf-compatible model-theoretic semantics. Technical report, W3C.

Salvadores, M., Horridge, M., Alexander, P. R., Fergerson, R. W., Musen, M. A., and Noy, N. F. (2012). Using sparql to query bioportal ontologies and metadata. In *The Semantic Web–ISWC 2012*, pages 180–195. Springer.

Salvadores, M., Alexander, P. R., Musen, M. A., and Noy, N. F. (2013). Bioportal as a dataset of linked biomedical ontologies and terminologies in rdf. *Semantic web*, **4**(3), 277–284.

Sazonau, V., Sattler, U., and Brown, G. (2013). Predicting performance of owl reasoners: Locally or globally? Technical report, Technical report, School of Computer Science, University of Manchester.

Seaborne, A. and Prud'hommeaux, E. (2008). SPARQL query language for RDF. W3C recommendation, W3C. http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.

Sioutos, N., de Coronado, S., Haber, M. W., Hartel, F. W., Shaiu, W.-L., and Wright, L. W. (2007). Nci thesaurus: a semantic model integrating cancer-related clinical and molecular information. *Journal of biomedical informatics*, **40**(1), 30–43.

Steigmiller, A., Liebig, T., and Glimm, B. (2014). Konclude: System description. *Web Semantics: Science, Services and Agents on the World Wide Web*, **27**(1).

The Uniprot Consortium (2007). The universal protein resource (uniprot). *Nucleic Acids Res*, **35**(Database issue).

Tobies, S. (2000). The complexity of reasoning with cardinality restrictions and nominals in expressive description logics. *J. Artif. Int. Res.*, **12**(1), 199–217.

Tudose, I., Hastings, J., Muthukrishnan, V., Owen, G., Turner, S., Dekker, A., Kale, N., Ennis, M., and Steinbeck, C. (2013). Ontoquery: easy-to-use web-based owl querying. *Bioinformatics*, **29**(22), 2955–2957.

Williams, A. J., Harland, L., Groth, P., Pettifer, S., Chichester, C., Willighagen, E. L., Evelo, C. T., Blomberg, N., Ecker, G., Goble, C., and Mons, B. (2012). Open phacts: semantic interoperability for drug discovery. *Drug Discovery Today*, **17**(2122), 1188 – 1198.

Xiang, Z., Mungall, C. J., Ruttenberg, A., and He, Y. (2011). Ontobee: A linked data server and browser for ontology terms. In *Proceedings of International Conference on Biomedical Ontology*, pages 279–281.

Disease Compass – A Navigation System for Disease Knowledge based on Ontology and Linked Data Techniques

Kouji Kozaki[1,*], Yuki Yamagata[1], Riichiro Mizoguchi[2,*],
Takeshi Imai[3] and Kazuhiko Ohe[3]
[1] ISIR, Osaka University, 8-1 Mihogaoka, Ibaraki, Osaka, Japan
[2] Research Center for Service Science School of Knowledge Science, Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, Ishikawa, Japan
[3] Department of Medical Informatics, Graduate School of Medicine, The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo, Japan

ABSTRACT

This paper discusses a navigation system for disease knowledge named *Disease Compass*. It navigates the users through a disease ontology defined based on River Flow Model of diseases which captures a disease as causal chains of abnormal states. The disease ontology is published using linked data techniques so that medical information systems can use it as knowledge infrastructure about disease with other related knowledge sources. Because the disease ontology has been developed under a tight collaboration between ontology engineers and medical experts, it could be a valuable knowledge base for advanced medical information systems. Furthermore, linked data techniques enable us to obtain related information from other linked data or web services. Based on these techniques, the users of *Disease Compass* can browse causal chains of a disease and obtain related information about the selected disease and abnormal states from the following two web services. One is general information from linked data such as DBpedia, and the other is a 3D image of anatomies. Such a functionality was enabled thanks to the disease ontology which is successfully combined with other web resources. As a result, *Disease Compass* can support the users to explore disease knowledge with related information from various point of views.

1 INTRODUCTION

Recently, medical information resources storing considerable amount of data are available. Semantic technologies are expected to contribute to the effective use of such information resources and many medical ontologies such as SNOMED-CT[1], OGMS (Scheuermann 2009) have been developed for realizing sophisticated medical information systems. Although medical ontologies consist of various domains such as diseases, anatomy, drug, clinical information etc., disease is an important concept because its complicated mechanisms are deeply related to other concepts across many of these medical domains.

This is why we focus on developing disease ontology. Some disease ontologies such as DOID (Osborne 2009), and IDO (Cowell 2010) have been developed. They mainly focus on the ontological definition of a disease with related properties. On the other hand, we proposed a definition of a disease that captures it as a causal chain of abnormal states and a computational model called the River Flow Model of a Disease (Mizoguchi 2011). Our disease ontology consists of rich information about causal chains related to each disease. The causal chains provide domain-specific knowledge about diseases, answering questions such as "What disorder/abnormal state causes a disease?" and "How might the disease advance, and what symptoms may appear?" We believe it could be a valuable knowledge base for advanced medical information systems.

This paper discusses a navigation system for disease knowledge named *Disease Compass* based on the disease ontology which we developed. The system has two special features. Firstly, its user can browse disease knowledge according to causal chains of diseases which are defined in the disease ontology. Secondly, the user can obtain related information about the selected disease based on linked data techniques. These functionalities of the *Disease Compass* can support the users to understand disease knowledge from various points of view.

This paper is organized as follows. In Section 2, we introduce our disease ontology. In Section 3, we outline how to publish the disease ontology as linked data. In Section 4, we discuss a navigation system for disease knowledge named *Disease Compass*. Finally, in Section 5, we present concluding remarks and a give an outline of future work.

2 A DISEASE ONTOLOGY

2.1 Definition of a Disease

A typical disease, as a dependent continuant, enacts extending, branching, and fading processes before it disappears. As a result of these processes, a disease can be identified as a continuant that is an enactor of those processes. Such an entity (a disease) can change according to its phase while maintaining its identity. On the basis of this observation, we defined a disease and related concepts as follows (Mizoguchi 2011).

Definition 1: A disease is a dependent continuant constituted of one or more causal chains of clinical disorders (abnormal state) appearing in a human body and initiated by at least one disorder.

When we collect individual causal chains belonging to a particular disease type (class), we can find a common causal

* To whom correspondence should be addressed: kozaki@ei.sanken.osaka-u.ac.jp and mizo@jaist.ac.jp

[1] http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html

chain (partial chain) that appears in all instance chains. By generalizing such a partial chain, we obtain the notion of a core causal chain of a disease as follows.

Definition 2: A core causal chain of a disease is a sub-chain of the causal chain of a disease, whose instances are included in all the individual chains of all instances of a particular disease type. It corresponds to the essential property of a disease type.

Definition 2 provides a necessary and sufficient condition for determining the disease type to which a given causal chain of clinical disorders belongs. That is, when an individual causal chain of clinical disorders includes instances of the core causal chain of a particular disease type, it belongs to that disease type. We can thus define such a disease type, which includes all possible variations of physical chains of clinical disorders observed for patients who contract the disease. According to a standard definition of subsumption, we can introduce an *is-a* relationship between diseases using the chain-inclusion relationship between causal chains.

Definition 3: *Is-a* relationship between diseases. Disease A is a supertype of disease B if the core causal chain of disease A is included in that of disease B. The inclusion of nodes (clinical disorders) is judged by taking an *is-a* relationship between the nodes, as well as sameness of the nodes, into account.

Definition 3 helps us systematically capture the necessary and sufficient conditions of a particular disease, which roughly corresponds to the so-called "main pathological conditions." Assume, for example, that (non-latent) diabetes and type-I diabetes are, respectively, defined as <*deficiency of insulin → elevated level of glucose in the blood*> and <*destruction of pancreatic beta cells → lack of insulin I in the blood → deficiency of insulin → elevated level of glucose in the blood*>. Then, we get <*type-I diabetes is-a (non-latent) diabetes*> according to Definition 3.

2.2 Types of causal chains in disease definitions

In this paper, we call causal chains that appear in the disease definition disease chains. In theory, we can consider three types of causal chains that appear in the disease definition, when we define a disease:

General Disease Chains are all possible causal chains of (abnormal) states in a human body. They can be referred to by any disease definition.

Core Causal Chain of a disease is a causal chain that appears in all patients of the disease.

Derived Causal Chains of a disease are causal chains obtained by tracing general disease chains upstream or downstream from the core causal chain. The up-stream chains imply possible causes of the disease, and the downstream ones imply possible symptoms in a patient suffering from the disease.

Fig.1 shows the main types of diabetes constituted by the corresponding types of causal chains. The figure shows that

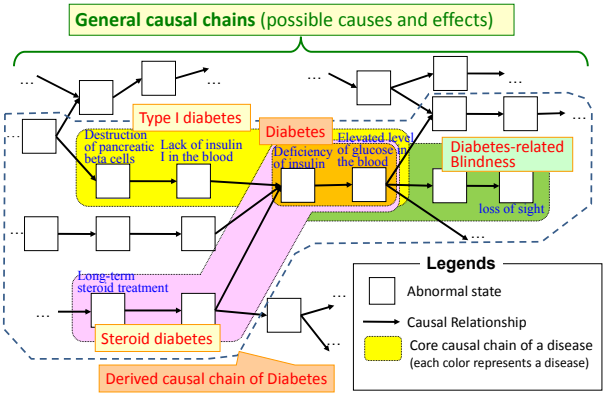


Fig. 1 Types of diabetes constituted of causal chains.

subtypes of diabetes are defined by extending its core causal chain according to its derived causal chains upstream or downstream.

Note here that it is obviously difficult to define all general causal chains in advance, because it is impossible to know all possible states in the human body and the causal relationships among them. In order to avoid this difficulty, we define the general disease chains by generalizing core/derived causal chains of every disease defined by clinicians in bottom-up approach. That is, we ask clinicians to define only core causal chains and typical derived causal chains of each disease, according to their knowledge and the textbooks on the disease. And then define the general disease chains by generalizing them.

Our disease ontology has been developed by clinicians in the 13 special fields. As of 11 May 2013, it has about 6,302 disease concepts and about 21,669 disorder (abnormal state) concepts with causal relationships among them.

3 DISEASE ONTOLOGY AS LINKED DATA

3.1 Basic policy to publish the disease ontologies as linked data

There are several approaches for system development based on ontologies. One of typical approaches is to use some APIs for ontology processing. Because our disease ontology is built using Hozo[2], we can develop application systems using APIs for Hozo ontologies. We can also use API for OWL since Hozo has OWL exporting function. On the other hand, linked data techniques are very efficient to develop applications across several datasets published on the Web. Because disease knowledge is related to various knowledge in other medical domains, we take an approach to publish the disease ontology as linked data to develop application system based on it.

[2] http://www.hozo.jp/

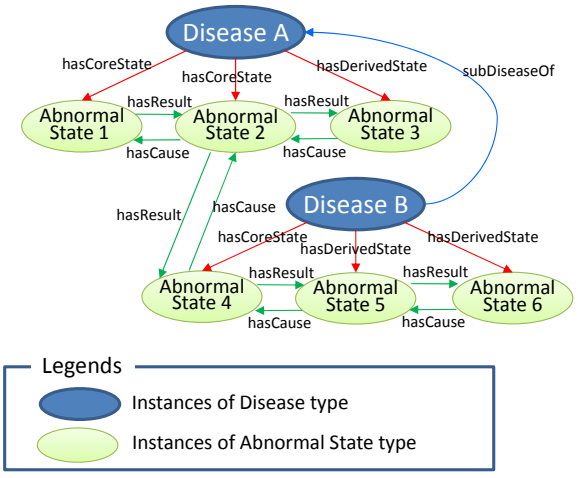


Fig.2 An RDF representation of diseaes.

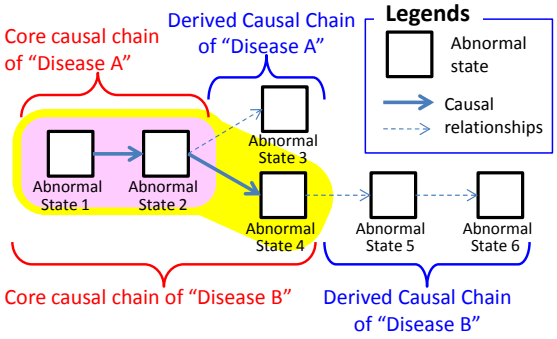


Fig.3 Causal chains of diseases shown in Fig.2.

Because the standard format for linked data is RDF, it may be regarded an easy task to publish ontologies in RDF formats using OWL or RDF(S) as linked data. However, an ontology language such as OWL is designed for mainly class descriptions, and the assumption is that the language will be used for reasoning based on logic; yet finding and tracing connections between instances are main tasks in linked data. Therefore, OWL/RDF and RDF(S) are not always convenient or efficient for linked data because of their complicated graph structures. This is problematic, especially when we want to use an ontology's conceptual structures as a knowledge base with rich semantics.

Consequently, we consider to design an RDF data model for publishing our disease ontology as linked data (Kozaki 2013). We outline the RDF model in the next section.

3.2 RDF Model for causal chains of diseases

After we constructed the disease ontology, we extracted information about causal chains of diseases from it and converted them into RDF formats as a linked data. We call the dataset *Disease Chain-LD*. It consists of diseases, abnormal states, and the relationships among them. Abnormal states

(1) Get all abnormal states.

```
select ?abn
where {?abn rdf:type dont:Abnormal_State}
```

(2) Get all cause of abnormal state <abn_id>.

```
select ?o
where {<abn_id> dont:hasCause* ?o }
```

(3) Get all causal chains which appear in definitions of disease <dis_id> as a list of abnormal state.

```
select DISTINCT ?o
where { <dis_id> dont:subDiseaseOf* ?dis .
 {?dis dont:hasCoreState ?o }
 UNION {?dis dont:hasDerivedState ?o }}
```

Fig.4 Example queries. In this figure "dont:" represents a prefix of the *Disease Chain-LD* and <abn_id> and <dis_id> represents id of a selected abnormal state and disease respectively.

are represented by instances of *Abnormal_State* type. Causal relationships between them are represented by describing *hasCause* and *hasResult* which are inverse properties. Abnormal states connected by these properties are a possible cause/result. Therefore, general disease chains can be obtained by collecting all abnormal states according to these connections.

Diseases are represented by instances of *Disease* type. Abnormal states that constitute a core causal chain and a derived causal chain of a disease are represented by *hasCoreState* and *hasDereivedState* properties, respectively. *Is-a (sub-class-of)* relationships between diseases and abnormal states are represented by *subDiseaseOf/subStateOf* properties instead of *rdfs:subClassOf* because diseases and abnormal states are represented as RDF resources, while *rdfs:subClassOf* is a property between *rdfs:Classes*.

Fig.2 shows an example of RDF representation of diseases. It represents *disease A* and its sub-disease *disease B*, whose causal chains are shown in Fig.3. Note that causal chains consist of abnormal states and causal relationships between them. Therefore, when we obtain a disease's core causal chain or derived causal chain, we have to obtain not only abnormal states connected to the disease by *hasCoreState/hasDereivedState* properties but also causal relationships between them. Although causal relationships are described without determining whether they are included in the causal chains of certain diseases, we can identify the difference by whether abnormal states at both ends of *hasCause/hasResult* properties are connected to the same disease by *hasCoreState/hasDereivedState* properties. Furthermore, when we obtain the causal chain of a disease that has a super disease, such as disease B in Fig.2, we have to obtain causal chains of its super disease in addition to the causal chain directly connected with it, and aggregate them.

The processing is not complicated; it just requires simple procedural reasoning. In summary, we can obtain a disease's causal chains, which define the disease through several SPARQL queries to the dataset. Fig.4 shows some

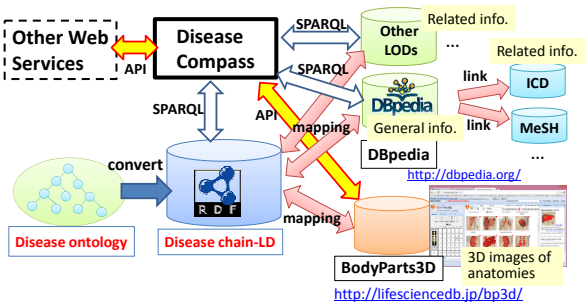


Fig.5 System architecture of Disease Compass.

example queries to obtain disease chains. We confirmed that we can obtain every information about disease chains using SPARQL queries (Kozaki 2013).

We published the disease ontology as linked data based on our RDF model. It includes definitions of 2,103 diseases and 13,910 abnormal states in six major clinical areas extracted from the disease ontology on May 11, 2013. The dataset contained 71,573 triples[3].

4 DEVELOPMENT OF DISEASE COMPASS

4.1 Disease Compass

It is not easy to use SPARQL for medical experts while every piece of information about disease chains can be obtained using SPARQL queries. Therefore, we developed a navigation system for disease knowledge named *Disease Compass*[4]. We designed the system so that the users can easily explore disease knowledge with related information even if they do not know about ontology or linked data.

4.2 System architecture

Fig.5 shows the system architecture of *Disease Compass*. The system obtains disease knowledge from Disease Chain-LD which is converted from the disease ontology. It has mapping information with other LODs (Linked Open Data) and web services. The system can obtain related information through these mappings. Though the system currently has mappings only to DBpedia and BodyPart3D, we can extend mappings to other LODs using existing approach to generate such linkages such as Song 2013).

Technically, the system uses two ways to access these datasets. One is SPARQL queries for linked data and the other is API for web services. If related resources (ontologies and other datasets) are published as LOD, the system is easily extended to link such related information using SPARQL. It is a large benefit to use linked data techniques. Please note that many linked data includes links to others. For example, DBpedia includes links to major medical codes such as ICD10 and MeSH. It means that the system can follows these links through mappings between Disease Chain-LD and DBpedia.

Disease Compass is developed as a web service that supports not only PCs but also tablets or smartphones. It is implemented using Virtuoso for its RDF database and HTML 5 for visualizations of disease chains and other information. All modules of the systems provides APIs for other web services. It enable others to use all functions of Disease Compass so that they work with related services.

4.3 User interfaces for navigation

Fig.6 shows user interface of *Disease Compass*. The users select a disease according to *is-a* hierarchy of diseases or search a disease chain by disease name or abnormal state which is included in it. The system visualizes disease chains of selected disease in a user friendly representation on the center of the window.

At the same time, the system obtains and shows related information about the selected disease and abnormal state from the following two web services. One is general information from linked data such as DBpedia, and the other is a 3D image of anatomies.

DBpedia[5] is a linked open dataset extracted from Wikipedia. It provides general information about diseases while its content is not authorized by medical experts. We suppose its contents is valuable enough to get an overview of diseases. In fact, it also gives links to major medical terminology and codes such as ICD10 and Mesh. The users can follow these links when they want to know more special information about the disease. This technology to obtain related information from other web resources (ontologies, medical codes, datasets etc.) through mappings is easy to apply to other linked data. We plan to extend the target linked data in the near future.

On the other hand, a 3D image of anatomies are generated using a web service named BodyPart3D/Anatomography (Mitsuhasi 2009). The target area of the image is decided by *Disease Compass* to combine all target of abnormal states appearing in the definition (causal chains) of the selected disease chain. Then, the system highlights a part of 3D image which is target of the selected abnormal state in the disease chains.

Such a functionality was enabled thanks to the disease ontology which is successfully combined with other web resources based on linked data technologies. As a result, *Disease Compass* can support the users to explore disease knowledge with related information through various web resources towards integrations of disease knowledge.

[3] Although the disease ontology includes definitions diseases in 13 clinical areas, we published parts of them that were well reviewed by clinicians. We will publish the latest version on the end of March, 2014. We provides a SPARQL endpoint to access the disease ontology at http://lodc.med-ontology.jp/ .

[4] Disease Compass is available at http://lodc.med-ontology.jp/ .

[5] We use DBpedia English (http://dbpedia.org) and Japanese (http://ja.dbpedia.org).

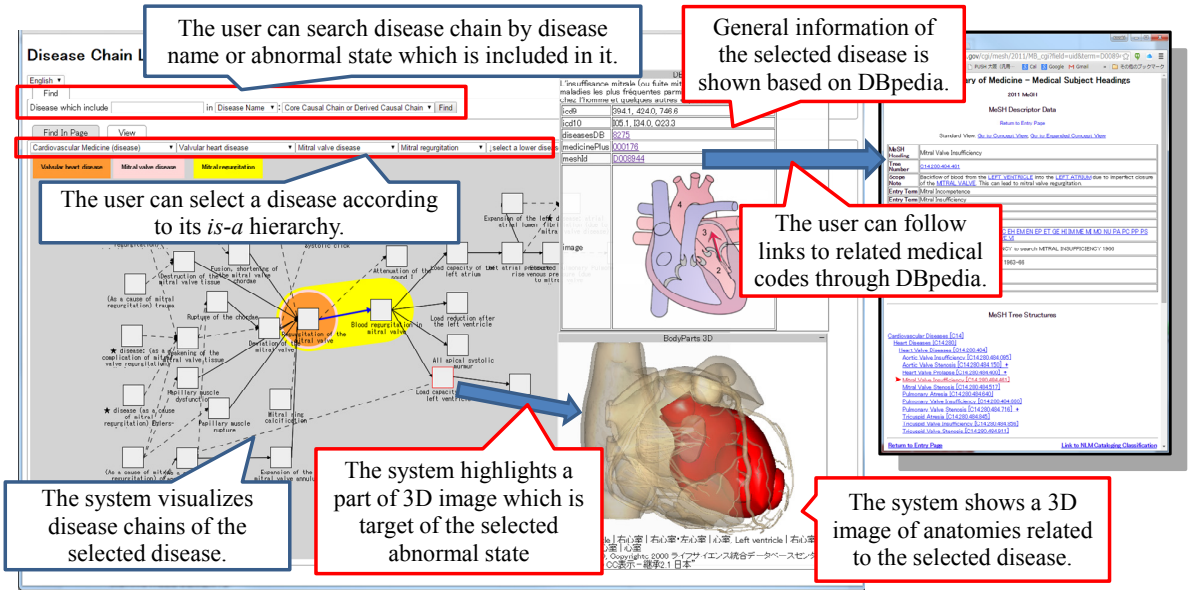


Fig.6 User interface of Disease Compass for navigation of disease knowledge.

We tried to extend the system towards integration of disease knowledge through ontologies and linked data technologies. As its first step, we investigated the differences in the hierarchical structure of biomedical resources and made a trial integration of our abnormality ontology and related resources such as PATO, HPO and MeSH based on ontological theory (Yamagata 2014).

As a result, we developed a prototype of the abnormality ontology as linked data with a browsing system. Thanks to mapping information with other resources, users can access disease knowledge through not only our abnormality ontology but also other open resources. We plan to extend this integration to our disease ontology and *Disease Compass*.

5 CONCLUDING REMARKS

This paper discusses a navigation system for disease knowledge based on the disease ontology and linked data technologies. Our disease ontology defines diseases based on causal chains of abnormal state (disorder) and a browsing system for it. It allows users to browse definitions of diseases with related information obtained from other linked data. We suppose that it can help them to understand about diseases from various point of views according to the users' interests and intention. The system was demonstrated for medical experts in some meetings and workshops and got positive comments while an evaluation by users is a future work.

Future work includes extension of related resources using linked data and developments of more practical applications using the Disease Chain LD. We also continue to improve the system including bug fixes and developments of new functions. The latest version of Disease Compass is available at the URL http://lodc.med-ontology.jp/ .

ACKNOWLEDGEMENTS

A part of this research is supported by the Japan Society for the Promotion of Science (JSPS) through its "FIRST Program"" and the Ministry of Health, Labour and Welfare, Japan. The authors are deeply grateful to Drs. Natsuko Ohtomo, Aki Hayashi, Takayoshi Matsumura, Ryota Sakurai, Satomi Terada, Kayo Waki, and other, at The University of Tokyo Hospital for describing disease ontology and assisting us with their broad clinical knowledge. We also would like to thank other team members, Drs. Yoshimasa Kawazoe, Masayuki Kajino, and Emiko Shinohara from The University of Tokyo for useful discussions related to biomedicine.

REFERENCES

Scheuermann, R. H., Ceusters, W., and Smith, B. (2009): Toward an Ontological Treatment of Disease and Diagnosis. *Proc. of the 2009 AMIA Summit on Translational Bioinformatics*, San Francisco, CA, 116-120.

Mizoguchi, R., Kozakil, K. and et al. (2011): River Flow Model of Diseases, *Proc. of ICBO2011*, Buffalo, USA, 63-70.

Osborne, J. D., et al. (2009): Annotating the human genome with Disease Ontology. *BMC Genomics* **10**(1):S6.

Cowell, L. G. and Smith, B (2010): Infectious Disease Ontology. *Infectious Disease Informatics*, Chapter 19, Sintchenko V., 373-395.

Dezhao Song and Jeff Heflin. (2013): Domain-Independent Entity Coreference for Linking Ontology Instances. *ACM Journal of Data and Information Quality* **4**(2), Article 7.

Mitsuhashi, N. et al. (2009): BodyParts3D: 3D structure database for anatomical concepts, *Nucleic Acids Research* **37**, D782-D785.

Kozakil, K., and et al.(2013): Publishing a Disease Ontologies as Linked Data, *Proc. of JIST2013*, Seoul, Korea, 110-128.

Yamagata Y, Kozaki K, Mizoguchi R, Imai T and Ohe K.(2014):Towards the Integration of Abnormality in Diseases, Proc. of ICBO 2014, 7-12.

Part II

Early Career Papers

Annotating biomedical ontology terms in electronic health records using crowd-sourcing

Andre Lamurias [1,2]*, Vasco Pedro [3], Luka Clarke [2] and Francisco M. Couto [2]

[1]BioISI: Biosystems & Integrative Sciences Institute, Faculdade de Ciências, Universidade de Lisboa, Portugal
[2]LaSIGE, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa, 1749-016, Lisboa, Portugal
[3]Unbabel, 360 3rd Street, Suite 700, San Francisco, CA 94107-1213, USA

ABSTRACT

Electronic health records have been adopted by many institutions and constitute an important source of biomedical information. Text mining methods can be applied to this type of information to automatically extract useful knowledge. We propose a crowd-sourcing pipeline to improve the precision of extraction and normalization of biomedical terms. Although crowd-sourcing has been applied in other fields, it has not been applied yet to the annotation of health records. We expect this pipeline to improve the precision of supervised machine learning classifiers, by letting the users suggest the boundaries of the terms, as well as the respective ontology concept. We intend to apply this pipeline to the recognition and normalization of disorder mentions (i.e., references to a disease or other health related conditions in a text) in electronic health records, as well as drug, gene and protein mentions.

1 INTRODUCTION

Electronic health records (EHRs) are a source of information relevant to various research areas of biomedicine. These records contain details on diseases, symptoms, drugs and mutations, as well as relations between these terms. As more institutions adopt this type of system, there is an increasing need for methods that automatically extract information from textual data. This information may be matched to existing ontologies, with the objective of either validating the information extracted or expand the ontology with new information.

Text mining methods have been proposed to automatically extract useful information from unstructured text, such as EHR. Named Entity Recognition (NER) is a text mining task which aims at identifying the segments of text that refer to an entity or term of interest. Another task is normalization, which consists of assigning an ontology concept identifier to the recognized term. Finally, the relations described between the identified terms can be extracted, which is known as Relation Extraction.

The results of these tasks should be as accurate as possible so that minimal human intervention is required to use the results for other applications. To evaluate fairly the state-of-the-art of text mining systems, community challenges have been organized, where the competing systems are evaluated on the same gold standard. The task 14 of SemEval 2015 consisted in the NER of disorder mentions from EHR, as well as the normalization to the SNOMED-CT subset of UMLS (Campbell *et al.*, 1998). The best F-measure obtained for this task was of 75.5%. The CHEMDNER task of BioCreative IV consisted in the recognition of chemical entities in the titles and abstracts of PubMed articles. For this task, the best F-measure was of 87.39%. The difference between the results of the two tasks could be due to the fact that EHR may contain more noise than scientific articles. These results show that there is a need to improve the state-of-the-art, to satisfy the user expectations on automated extraction of biomedical information from unstructured text.

In this paper we propose a pipeline to improve the extraction and normalization of biomedical ontology terms in EHR by crowd-sourcing the validation of the results obtained with machine learning algorithms. This approach has been applied to other types of tasks, with promising results. The crowd would be used to validate the boundaries of the term, as well as the ontology concept associated.

2 NORMALIZATION OF BIOMEDICAL TERMS TO ONTOLOGIES

The results produced by NER methods may be normalized to unique identifiers from ontologies. The advantage of this approach is that the structure of the reference ontology may be used to validate the information extracted from the text. We have explored semantic similarity between chemical entities matched to ChEBI concepts, which improved the precision of our system (Lamurias *et al.*, 2015).

The normalization of entities is a challenge due to the ambiguity and variability of the terminology. The same label may refer to different concepts, depending on the context, while one concept may be mentioned with different names, due to spelling variants, abbreviations and capitalization. While the ontology may provide a set of synonyms for each concept, it is usually incomplete, requiring a method more advanced than string matching to correctly normalize an entity.

3 CROWD-SOURCING IN ANNOTATION TASKS

Text processing tasks are suitable candidates for crowd-sourcing since they cannot be solved computationally, and can be broken down into smaller micro-tasks (Good and Su, 2013). For example, it has been applied to machine translation (Ambati and Vogel, 2010), recognition of names in historical records (Sukharev *et al.*, 2014), question-answering (Mrozinski *et al.*, 2008) and ontology alignment (Sarasua *et al.*, 2012). Crowd-sourcing micro-tasks are usually defined by the large volume of tasks to be performed, as well as the simplicity of each individual task. The participants may be motivated by monetary rewards (e.g. Amazon Mechanical Turk), games with purpose (Von Ahn and Dabbish, 2008), or simply the satisfaction of having contributed to a larger project (Jansen *et al.*, 2014).

*To whom correspondence should be addressed: alamurias@lasige.di.fc.ul.pt

Computational methods to map a term to an ontology concept, usually based on string similarity, are able to find one or more matches for each term. However, a machine is not able to identify the most correct term from a list of matches with the accuracy of a human annotator. By letting a large number of participants evaluate the ontology concepts matched to the terms recognized in a given text, a new dataset can be generated with these corrections. This dataset would be used to train a classifier able to determine the correct concept corresponding to a recognized biomedical concept, such as disorder, chemical, protein or gene, with high precision. This classifier can be trained with a supervised machine learning algorithm, or with reinforcement learning. Likewise, a golden dataset could be generated to evaluate and tune the classifier.

4 PIPELINE

The pipeline is composed by two modules: one for NER of disorder, chemical, gene and protein mentions, and another for normalization to SNOMED-CT, ChEBI and Gene Ontology concepts, respectively.

The NER modules starts with classifiers trained with existing annotated corpora. We have trained classifiers based on the Conditional Random Fields algorithm (Lafferty *et al.*, 2001) for both disorder and chemical entity mentions. We will train more classifiers to recognize gene and protein mentions, with existing corpora annotated with those types of entities. The results of these classifiers will be evaluated by the crowd, who will be able to accept the entity and its boundaries, adjust the boundaries, or reject the entity if it does not correspond at all to what the classifier predicted. These corrections will be used to improve the performance of the first step, through reinforcement learning, with different weights assigned to the specialists according to their usage profile.

The normalization module will first attempt to map the string to a concept of the respective ontology. Since multiple matches may exist for the same string, this ambiguity will be solved with a semantic similarity measure. These mappings will be evaluated by the crowd, why the option of accepting the concept as correct, or choosing another one from the same ontology. As before, these corrections will be used to train a machine learning classifier, using the semantic similarity values as features.

For example, taking the sentence "The rhythm appears to be atrial fibrillation" as input, the NER classifier may recognize only the word "fibrillation" as a disorder mention. In this case, the boundary of the term may be extended to include "atrial". In SNOMED-CT, several concept are related to atrial fibrillation, for example, "Atrial fibrillation" (C0004238) and "Atrial fibrillation and flutter" (C0155709). If the second concept is chosen by the system instead of the first one, the user may indicate this mistake. Otherwise, the user will confirm that the mapping is correct.

Every document processed by our system is anonymized using standard procedures, which includes removing all references to personal details. The user only evaluates individual phrases containing annotations, to prevent the re-identification of documents. We will apply a sliding-window approach to harmonize the evaluations performed by the crowd, so that each phrase evaluated by a user should overlap with other phrases. With this strategy, we can align the sequence of phrases that was accepted by the majority of the crowd, and prevent errors committed due to the lack of context.

As an incentive to the participation of users, we intend to apply a mechanism of rewards based on a virtual currency. KnowledgeCoin (Couto, 2014) is a virtual currency that was originally proposed to reward and recognize data sharing and integration on the semantic web. This could also be applied to the proposed pipeline, by distributing KnowledgeCoins for each text validated by a user, improving the reputation of that user. Potential participants in this kind of project would be medicine students. The University of Lisbon accepts almost three hundred medicine students per year, which could provide a relatively large crowd for our pipeline. Retired physicians, nurses, physician assistants and researchers may also participate, in order to provide more specialized curation. This type of crowd has been used by CrowdMed to provide crowd-sourced diagnostics to complex medical cases, with high levels of accuracy.

5 CONCLUSION

We propose a novel pipeline for recognition and normalization of biomedical terms to ontology concepts, using crowd-sourcing. The complete and automatic annotation of biomedical texts such as EHR requires systems with high precision. The normalization task is particularly challenging due to the subjective nature of ontology mapping. By letting a large group of specialized participants correct the mistakes of a machine learning classifier, we expect an improvement of the performance of current biomedical text mining systems. The idea is not only to create a scalable knowledge base but help from a community of specialist curators that may be available to help in creating a golden standard for a new biomedical area, or improve current results, or just validate some results.

ACKNOWLEDGEMENTS

This work was supported by the Fundação para a Ciência e a Tecnologia (https://www.fct.mctes.pt/) through the PhD grant PD/BD/106083/2015, the Biosys PhD programme and LaSIGE Unit Strategic Project, ref. PEst-OE/EEI/UI0408/2014.

REFERENCES

Ambati, V. and Vogel, S. (2010). Can crowds build parallel corpora for machine translation systems? In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 62–65. Association for Computational Linguistics.

Campbell, K. E., Oliver, D. E., and Shortliffe, E. H. (1998). The unified medical language system toward a collaborative approach for solving terminologic problems. *Journal of the American Medical Informatics Association*, **5**(1), 12–16.

Good, B. M. and Su, A. I. (2013). Crowdsourcing for bioinformatics. *Bioinformatics*, page btt333.

Jansen, D., Alcala, A., and Guzman, F. (2014). Amara: A sustainable, global solution for accessibility, powered by communities of volunteers. In *Universal Access in Human-Computer Interaction. Design for All and Accessibility Practice*, pages 401–411. Springer.

Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Lamurias, A., Ferreira, J. D., and Couto, F. M. (2015). Improving chemical entity recognition through h-index based semantic similarity. *Journal of Cheminformatics*, **7**(Suppl 1), S13.

Mrozinski, J., Whittaker, E., and Furui, S. (2008). Collecting a why-question corpus for development and evaluation of an automatic qa-system. In *46th Annual Meeting of the Association of Computational Linguistics: Human Language Technologies*, pages 443–451.

Sarasua, C., Simperl, E., and Noy, N. F. (2012). Crowdmap: Crowdsourcing ontology alignment with microtasks. In *The Semantic Web–ISWC 2012*, pages 525–541. Springer.

Sukharev, J., Zhukov, L., and Popescul, A. (2014). Learning alternative name spellings on historical records.

Von Ahn, L. and Dabbish, L. (2008). Designing games with a purpose. *Communications of the ACM*, **51**(8), 58–67.

Replacing EHR structured data with explicit representations

Jonathan Bona[1] and Werner Ceusters[2]
[1] Department of Oral Diagnostic Sciences, University at Buffalo, 355 Squire Hall, Buffalo, USA
[2] Department of Biomedical Informatics, University at Buffalo, 921 Main street, Buffalo, USA

1 INTRODUCTION

As part of a project to develop a roadmap for the creation of a multi-center fully identified patient data warehouse involving all State Universities of New York State (SUNY), we've examined patient records stored in an EHR database to 1) determine what its contents are intended to represent, and 2) develop ontologically sound models based on the principles of Ontological Realism and Referent Tracking (Ceusters, Chiun Yu Hsu, & Smith, 2014; Smith & Ceusters, 2010). The exploration of the EHR database is driven by identifying the structures that contain answers to questions that might be obtained with relative ease using the EHR system's user interface but that are difficult to find by working directly with the database, for example: '*what diagnoses have been made about which disorders a specific patient is suffering from; when were those diagnoses made and by whom; what entities are those diagnoses about?*'

This abstract presents issues with the data model currently used in the EHR database and an approach to address them.

2 CHARACTERIZING DATA AND ISSUES

The EHR database presents several obstacles to properly understanding its contents. The intended meaning of its data elements is not explicitly specified, but implicitly depends on connections to the user interface, other software that uses it, workflows, etc. Nevertheless, it's possible to determine some of this by examining tables, their elements and the connections between them.

For example, the tables named `Person` and `Problem` are linked to healthcare processes. `Problem` entries are organized under `Problem Headers`, where each header entry is supposed to correspond to a single thing (diagnosis, procedure) and `Problem` entries are spread out in time each under its header and correspond to updates to the record made during encounters (Weed, 1968). By focusing on patterns of diagnoses stored in these tables, we have identified several ways in which the data fail to represent. These include: multiple entries standing for the same entity; single entries that stand for more than one entity; entries that might represent either more than one entity of the same type separated in time, or a single entity that persists over time; entries wrongly marked as *resolved* or *errors*; and wrong or outdated active entries.

3 EXAMPLE

Figure 1 shows selected `Problem` entries and their `Problem Headers` for four patients. Within an example each row shows the problems under a single header for the patient. Columns represent days with entries for the patient. A filled oval indicates a problem entry on that day. This figure shows ordering of updates but not their spacing in time. A table of header descriptions appears on the next page.

In **example 1**, two headers are created initially: **e1ph1**

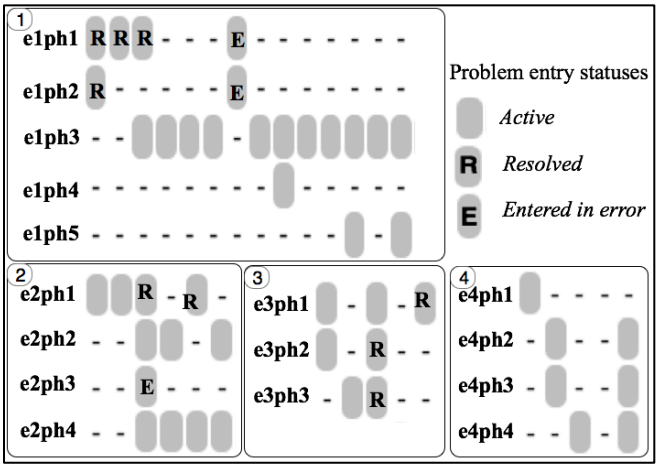


Figure 1 Examples of Problem entries for four patients

(*Diabetes mellitus type II*) and **e1ph2** (*Diabetes Mellitus With Complication*). Later **e1ph1** gets a new entry. Two months later, **e1ph3** (*Diabetes Mellitus*) is created. It is updated regularly. Six months after their creation, **e1ph1** and **e1ph2** are updated with new entries with the status *Error*. Later, **e1ph4** (*Type 1 Diabetes Mellitus - Uncontrolled*) and **e1ph5** (Diabetes *Mellitus With Complication*) are added. After that **e1ph3**, **e1ph4**, and **e1ph5** are updated occasionally, keeping the status *Active*.

The likely sequence of events is that the diagnosis of Type II DM was changed to Type I DM, after which a different header for Type I was created and used, and entries in the first two headers were marked *Error*. Old entries under those headers were marked as *Resolved*.

Clearly, this record has drifted from representing reality after just a few updates. This patient does not have more than one DM, but what was first thought to be Type I was then recognized as Type II. The system in this case fails to

distinguish 1) information that was retracted due to some change in knowledge about the patient's health from 2) information that was truly entered in error.

Example 2 shows a more correct use of the *Error* status: the header **e2ph3** (*Type II DM – Uncomplicated, Uncontrolled*) was created at the same time as headers about *ketoacidosis* (**e2ph2**) and *acanthosis nigricans* (**e2ph4**), which are complications of Diabetes. This mistake was quickly caught and **e2ph3** was marked *Error*. **e2ph1** here repeats the misuse of the *Resolved* status: the patient was thought to have Type I DM and the diagnosis changed in time. It's not that the Type I DM existed and then stopped existing.

Another issue is related to the use of ICD9 codes and ICD9-like descriptions: some Problem entries refer to multiple things. The patient's *Type II Diabetes* is one thing in the world; their *Ketoacidosis* is a separate related disorder.

A better representation would have identifiers for both the patient's diabetes and the patient's ketoacidosis, and would explicitly represent the relations between them (one was *caused by* or was a *complication of* the other).

e1ph1: Diabetes mellitus type II (NIDDM)
e1ph2: Diabetes Mellitus With Complication
e1ph3: Diabetes mellitus
e1ph4: DM (diabetes mellitus), type 1, uncontrolled
e1ph5: Diabetes mellitus with complication
e2ph1: Type 1 Diabetes Mellitus - Uncontrolled
e2ph2: Type II diabetes mellitus with ketoacidosis
e2ph3: Type 2 Diabetes Mellitus - Uncomplicated, Uncontrolled
e2ph4: Acanthosis nigricans
e3ph1: Closed Fracture Of The Shaft Of The Humerus
e3ph2: Closed Fracture Of Neck Of Femur - Transcervical
e3ph3: Closed Fracture Of The Humerus
e4ph1: Open Treatment Of Humeral Shaft Fracture w Plate/Screws
e4ph3: Fracture of left humerus

Example 3 shows a pattern of Problems for a patient who has multiple fractures. Entries in **e3ph1** and **e3ph2** indicate that the patient has closed fractures both of the shaft of the humerus, and of the neck of the femur. **e3ph3** (*Closed Fracture of the Humerus*) is created the following month. How many fractures are represented here: two or three? Probably the patient had two fractures: one in the femur and one in the humerus, and both **e3ph1** and **e3ph3** correspond to the same fracture. But possibly the patient had two fractures in the humerus at the same time, one specifically in the shaft and one in an unspecified place.

Example 4 shows a patient being treated for a humeral shaft fracture but there is initially no entry that represents the fracture itself – only for a treatment. **e4ph3** (*Fracture of left humerus*), which is created eighteen months after **e4ph1**, seems to represent the fracture itself. The next entry in **e4ph3** is *more than five years later*. Note that all Problems shown here have the status *Active*. None of these - including a six-year-old fracture - have been marked as *Resolved*.

4 EXPLICIT REPRESENTATIONS

We propose using ontology-based models that explicitly represent the entities and processes relevant to health care encounters. Figure 2 depicts an instance of *diagnosing a fracture* as part of a health care encounter.

Only some of the entities represented here will appear in other encounters. Any other diagnostic process will be a different instance. Its output will be a different instance than **diagnosis1** -- even if it's about the same things. The instance **diagnosis1** persists, even if it is later outdated, contradicted, or otherwise known to be wrong and marked as such. We also must be able to say that **disorder1** exists at time **t1** but not at time **t2**; that when it exists it is located at a particular spot on **bone1**; and that further fractures of **bone1** once **disorder1** has already healed are new fractures

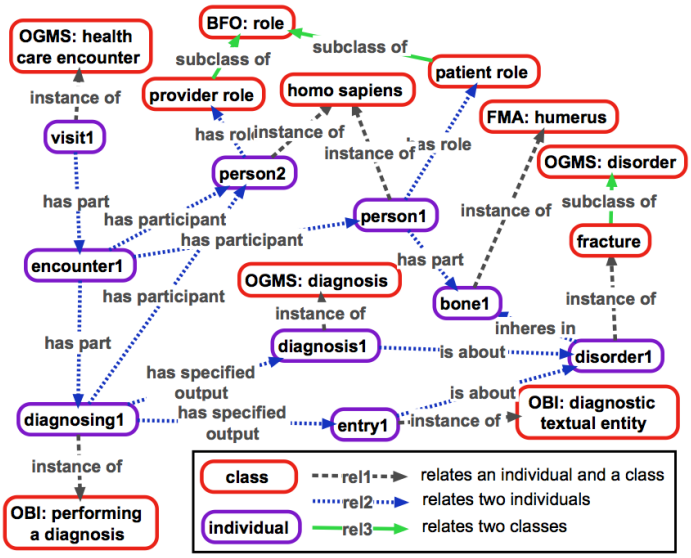


Figure 2 Representation of a fracture diagnosis

(**disorder2**). Having single identifiers for single entities (*this patient's diabetes*), which we can then say things about (*it is complicated by peripheral neuropathy*) is just as important as having separate terms for separate fractures.

Work is ongoing to develop computationally useful representations in OWL, mechanisms to interpret and translate patient data, and techniques to deal with temporal considerations and other issues that are not straightforward.

REFERENCES

Ceusters, W., Chiun Yu Hsu, & Smith, B. (2014). Clinical data wrangling using ontological realism and referent tracking. *CEUR Workshop Proceedings, 1237*, 27-32.

Smith, B., & Ceusters, W. (2010). Ontological realism: A methodology for coordinated evolution of scientific ontologies. *Applied Ontology, 5*(3-4), 139-188. doi: Doi 10.3233/Ao-2010-0079

Weed, L. (1968). Medical records that guide and teach. *New England Journal of Medicine, 278*, 593-600.

Compound Matching of Biomedical Ontologies

Daniela Oliveira* and Catia Pesquita

LaSIGE, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa, Campo Grande 1749-016, Portugal

ABSTRACT

Biomedical ontologies are particularly successful in the uniformization of the life sciences domain and ontology matching systems are useful to discover relationships between concepts of two different ontologies. However, that is also a limitation as there is a growing interest in discovering more complex kinds of mappings and existing techniques are limited to matching two ontologies. Therefore, producing 'compound' alignments, which match more than two ontologies, could be potentially useful to support a next generation of semantic technologies.

In this paper, we present a novel algorithm that produces compound matches between three different ontologies and its performance is evaluated against seven automatically inferred reference alignments from the biomedical domain. We analyze all alignments manually to verify the results and propose a new way to complete the logical definitions of OBO cross-products.

1 INTRODUCTION

Biomedical ontologies typically contain a high number of classes and many times cover the same field or related fields, which hinders their interoperability. One approach to address this problem is the use of matching systems which are capable of establishing meaningful connections between ontologies.

Still, most ontology matching systems produce equivalence mappings between classes or properties in two ontologies. However, in a complex domain such as biomedicine, where several ontologies describe different but related aspects of biomedical phenomena, it may be advantageous to create mappings by combining entities from more than two ontologies. We argue that it would be useful for the developers of ontology alignment systems to develop new techniques and tools for identifying 'compound matches', i.e. matches between class or property expressions involving more than two ontologies. To the best of our knowledge, there are currently no ontology matching systems capable of generating such mappings.

The purpose of this work is to develop novel algorithms which can be used for the efficient and effective creation of alignments between a class A of one ontology with an expression relating classes B and C of two other ontologies, constituting a ternary relationship.

2 METHODS

We consider that a ternary compound alignment is a set of correspondences (mappings) between classes from a source ontology O_s and class expressions obtained by combining two other classes each belonging to a different target ontology O_{t1} and O_{t2} (see Figure 1). This means that we define a ternary compound mapping as a tuple <X, Y, Z, R, M>, where X, Y and Z are classes from three distinct ontologies, R is a relation established between Y and Z to generate a class expression that is mapped to X via a mapping relation M. Here, we consider the ontology to which X belongs to be the source ontology, and the ontologies that define Y and Z to be the target ontology 1 and 2, respectively. In this particular case the relation R is always an intersection (regardless of any qualifier) and the mapping M an equivalence.

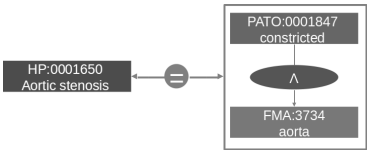


Fig. 1. Example of a possible ternary compound match.

2.1 Implementation

We developed a novel algorithm to establish compound mappings integrated into the AgreementMakerLight (AML) (Faria *et al.*, 2014) ontology matching system[1]. Our algorithm exploits AML's *Word Lexicon*, the set of all words in an ontology's vocabulary to which are assigned an evidence content (EC), reflecting the usage of the word within the ontology.

In a first step, we perform a pairwise mapping of the labels of O_s with the labels of O_{t1}, by the ratio of the sum of the EC of the words shared by the source label (l_s) and the target 1 label (l_{t1}), and the sum of the EC of the words in l_{t1}.

$$sim(l_s, l_{t1}) = \frac{\sum EC(word \in (l_s \cap l_{t1}))}{\sum EC(word \in l_{t1})} \quad (1)$$

We filter out all mappings with similarity below a given threshold. In a second step, for each mapping found in step 1, we remove from the source labels all the words that have already been matched (l_{s*}). Taking as an example the mapping in Figure 1, after matching HP and FMA, which would capture the mapping for 'aorta', the HP's class label would be reduced to 'stenosis'.

In a third step, for each mapping, we perform a pairwise comparison of the reduced source labels with target 2 labels. However, here the ratio divisor corresponds to the sum of EC of the words in the label with more words, to ensure the longest possible match.

$$sim(l_s, l_{t2}) = \frac{\sum EC(word \in (l_{s*} \cap l_{t2}))}{\sum EC(word \in longest(l_s, l_{t2}))} \quad (2)$$

In a fourth step, the final similarity between the matched labels is computed as the average between the similarities computed in steps 1 and 3. Label mappings below the second threshold are filtered out. Finally, the algorithm has a greedy selection step, which selects the

*To whom correspondence should be addressed: doliveira@lasige.di.fc.ul.pt

[1] Available at: https://github.com/AgreementMakerLight/AML-Compound

mapping with the highest similarity, amongst the source classes with more than one mapping.

2.2 Evaluation

To evaluate our strategy we used a set of seven reference alignments (Pesquita *et al.*, 2014) automatically created by inferring compound mappings from cross-products (Mungall *et al.*, 2011) of the logical definitions in OBO ontologies (Smith *et al.*, 2007). For this, we computed precision, recall and f-measure. We also performed a manual evaluation of the results, where we classified mappings into three possible categories: 'Correct', where the mapping is deemed correct and the source class has no mapping in the reference alignment; 'Conflict', where the mapping is deemed correct but the source class has a different mapping in the reference alignment; and 'Incorrect', where the mapping is deemed incorrect. We applied this to all mappings created by using 0.5 as a threshold for step 1 and 0.9 for step 2.

3 RESULTS

Table 1 presents some statistics about the alignments obtained. Preliminary results using this evaluation approach present low F-Measure, with a higher precision, which fluctuates between 67.9 and 11.6 and recalls that always fall below the 50% mark.

	Precision	Recall	F-Measure
MP-CL-PATO	52.6 %	20.8 %	29.8 %
MP-GO-PATO	67.9 %	47.2 %	55.7 %
MP-NBO-PATO	47.3 %	30.1 %	36.8 %
MP-UBERON-PATO	64.7 %	19.4 %	29.9 %
WBP-GO-PATO	11.6 %	7.7 %	9.2 %
HP-FMA-PATO	21.2 %	12.4 %	15.6 %

Table 1. Evaluation results from the comparison with the automated reference alignments

	Correct	Conflict	Incorrect
MP-CL-PATO	63.71 %	34.60 %	1.69 %
MP-GO-PATO	92.16 %	6.97 %	0.87 %
MP-NBO-PATO	72.46 %	26.09 %	1.45 %
MP-UBERON-PATO	91.33 %	7.96 %	0.70 %
WBP-GO-PATO	88.55 %	7.49 %	3.96 %
HP-FMA-PATO	77.82 %	15.56 %	6.61 %

Table 2. Manual evaluation of results.

The manual inspection of the mappings (Table 2) revealed that the algorithm is finding mostly correct mappings, with the lowest percentage belonging to the MP-CL-PATO compound alignment, which had the highest number of conflicting mappings.

4 DISCUSSION

One challenge in computing compound alignments is the memory requirements involved in the process. If matching two large biomedical ontologies is already a challenge for many ontology matching systems, handling three ontologies in a compound alignment scenario is even more demanding. Our algorithm reduces the search-space by using the two-step matching approach, which both reduces the time and memory requirements [2].

[2] The largest alignment takes less than 15 minutes with an Intel® Core™i7-2600 CPU 3.40GHz x 8 processor and 16GB memory.

Although our algorithm's performance against the reference alignments is low (Table 1), the manual evaluations of the mappings reveals a very low proportion of incorrect mappings, so we investigated how these new mappings could impact the logical definitions of the source ontology. The results presented in Table 3 indicate that the logical definitions of the three source ontologies could be expanded with more than 800 new logical definitions.

Ontology	New Mappings	OBO classes	% of Growth
MP	422	7694	5.48
WBP	182	957	19.02
HP	259	14059	1.84

Table 3. Influence of the new mappings on the source ontology.

We can conclude that our approach is capable of producing good precision (Table 2 shows an average of 81% of the matches are correct), and is able to find many correct mappings that are not in the reference alignment. However, it struggles with capturing many of the mappings in the references, which is mainly due to our algorithm's inability to distinguish between similar PATO class (e.g., PATO:0000470: 'present in greater numbers in organism' *vs.* PATO:0002002: 'has extra parts of type'), or the use of synonyms not defined in any of the ontologies.

5 CONCLUSION

We have presented, to the best of our knowledge, the first algorithm for compound matching of ontologies. It is particularly suited for biomedical ontologies, given its ability to handle large ontologies and the need in this domain to reveal more complex relations between them. Our preliminary experiments have shown that, despite the challenges in handling an increased matching space and the inherently more difficult-to-compute ternary mapping, our algorithm is able to produce good precision mappings. Moreover, we posit that it could also be used as a first step in adding new logical definitions to ontologies, since we were able to find several correct mappings that were not in the reference alignments..

ACKNOWLEDGEMENTS

The authors are grateful to Daniel Faria for his technical support. This work was supported by FCT through funding of LaSIGE Research Unit, ref.UID/CEC/00408/2013

REFERENCES

Faria, D., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2014). AgreementMakerLight: a scalable automated ontology matching system. *10th International Conference on Data Integration in the Life Sciences 2014 (DILS)*, page 29.

Mungall, C. J., Bada, M., Berardini, T. Z., Deegan, J., Ireland, A., Harris, M. A., Hill, D. P., and Lomax, J. (2011). Cross-product extensions of the Gene Ontology. *Journal of Biomedical Informatics*, **44**(1), 80 – 86. Ontologies for Clinical and Translational Research.

Pesquita, C., Cheatham, M., Faria, D., Barros, J., Santos, E., and Couto, F. M. (2014). Building reference alignments for compound matching of multiple ontologies using OBO cross-products. In *Ontology Matching Workshop at ISWC 2014*.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., *et al.* (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, **25**(11), 1251–1255.

Towards visualizing the mapping incoherences in Bioportal

Catarina Martins [1], Ernesto Jimenez-Ruiz [2], Emanuel Santos [1] and Catia Pesquita [1]

[1]LaSIGE, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa
[2]Department of Computer Science, University of Oxford

ABSTRACT

The integration of biomedical ontologies via their matching often results in logical errors, due to incorrect mappings or incompatible ontological models. To solve this issue, repair algorithms remove mappings, decreasing the size and possibly the coverage of the alignment. Understanding these errors is crucial to support an effective use of alignments, since different scenarios may favor coherence above completeness and vice-versa. This could be supported by visualization, however, the challenges in visualizing ontology alignments are further compounded by the need to provide an explanation to the logical conflict between the mappings.

We present a preliminary visualization tool that supports the identification of mapping incoherence, by displaying sets of mappings involved in logical conflicts between several BioPortal ontologies pairs, as well as the classes and axioms involved.

1 INTRODUCTION

Establishing meaningful correspondences between biomedical ontologies is crucial to effectively explore the knowledge they model in an articulated fashion. Creating these correspondences, or mappings, can be accomplished by ontology matching techniques (Euzenat *et al.*, 2007).

Bioportal (Whetzel *et al.*, 2011), a web portal that provides access to more than 400 biomedical ontologies, provides mappings between ontologies which are automatically generated or manually added by experts. However, the integration of the ontologies via these mappings can result in incoherences due either to erroneous mappings or incompatibilities between both ontologies (Meilicke and Stuckenschmidt, 2008). The example in Figure 1 illustrates this problem. Although individual mappings appear to be correct, their integration results in a logical conflict, since in NCI Thesaurus *Anatomic_Structure_System_or_Substance* is disjoint with *Gene_Product*, and *Fibrillar_Actin* cannot be a subclass of both. This is a result of the different domain models followed by Foundational Model of Anatomy (FMA) and National Cancer Institute Thesaurus (NCIT).

To address this issue, (Faria *et al.*, 2014) applied both AML (Faria *et al.*, 2013) and LogMap (Jiménez-Ruiz and Grau, 2011) to detect and repair the incoherences in 19 pairs of ontologies from Bioportal and their mappings, and discovered that 11 in 19 had logical errors involving in average 22% of the mappings. These algorithms aim at eliminating incoherences by removing or altering mappings, and although they can provide logically sound solutions, these may not always be correct, since the choice of which mappings to eliminate is based on a change minimization strategy.

In this context, a visualization tool to identify the incoherences caused by mappings between ontologies would support their identification and correction by expert users. Moreover, it would support the decision for or against using a repaired alignment, since coherence often sacrifices completeness, and depending on the application, users may prefer one or the other (Pesquita *et al.*, 2013). Here, we present a preliminary version of a web tool[1] for visualizing sets of mappings involved in causing incoherences between several BioPortal ontologies pairs, as detected by AML using its repair algorithm (Santos *et al.*, 2013).

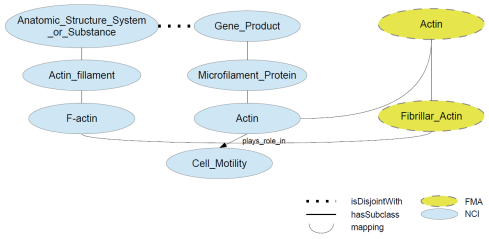


Fig. 1. Example of incoherent mappings between the FMA ontology and the NCIT
Taken with permission from Pesquita *et al.*, 2013

2 CHALLENGES IN VISUALIZING MAPPING INCOHERENCES

Biomedical ontologies present several visualization challenges due to their size, richness and complexity of vocabulary. There are two main paradigms to support the visualization of the ontologies and both have their drawbacks and benefits in the context of mapping visualization (Fu *et al.*, 2013). On the one hand, trees are appropriate when representing hierarchical relations but are confusing when representing multiple inheritance or several kinds of relations. Graphs, on the other hand, can handle both issues, but when the number of nodes is very high, visualization can be impaired. These aspects are relevant when visualizing alignments, and are further compounded by the representation of mappings between large ontologies (Pesquita *et al.*, 2014; Ivanova and Lambrix, 2014).

On top of the challenges in visualizing ontology alignments, there are additional constraints when considering the visualization of mapping incoherence. The goal here is to give the user sufficient information to understand the reason behind the incoherence and allow him or her to evaluate the mappings and decided if a correction is needed, and how it should be performed.

To accomplish this we have identified the minimum set of information to show when displaying a set of conflicting mappings: (1) the classes involved in the mappings; (2) the mappings between classes; (3) the disjoint axiom involved in causing the incoherence; and (4) the relations between the mapped classes and the classes involved in the disjoint axiom.

[1] http://xldb.di.fc.ul.pt/biotools/vizrepair/

Ontology 1	Ontology 2	Total Mappings	Conflicting Mappings
BDO	NCIT	1636	1374
CCONT	NCIT	2097	1136
EFO	NCIT	2507	1541
EP	FMA	78489	109
EP	NCIT	2465	307
MA	FMA	961	22
OMIM	NCIT	5178	1078
UBERON	FMA	1932	121

Table 1. Total and conflicting mappings in the ontologies used.Bone Dysplasia Ontology (BDO), Cell Culture Ontology (CCONT), Experimental Factor Ontology (EFO), Cardiac Electrophysiology Ontology (EP), Foundational Model of Anatomy (FMA), Mouse Adult Gross Anatomy Ontology (MA), National Cancer Institute Thesaurus (NCIT), Online Mendelian Inheritance in Man (OMIM), Uber Anatomy Ontology (UBERON).

Our preliminary webtool represents this complexity by using graph-based visualization techniques. However, given the drawbacks of graphs when displaying a considerable number of nodes, our webtool focuses on displaying sets of conflicting mappings (i.e., the mappings that taken together cause an incoherence), rather than the whole alignment at once.

3 WEBTOOL

The backend of our tool is supported by a database that stores all the ontology and alignment information. This corresponds to the ontologies and alignments shown in Table 1, as well as a list of the conflict sets. Having data stored in a relational database allows for faster retrieval of the information to draw the graph.

The user interface allows users to select an alignment, and then browse the conflict sets in a table format. Each conflict set can be selected for graph visualization (supported by sigma.js). To allow users to understand the conflict we need to show the relations between the classes involved in mappings and the classes involved in the disjoint axiom(s), however in many cases this would result in showing the several classes that compose the path from the mapping to the disjoint axiom. To reduce this visual clutter, we compute the transitive closure between these classes and display them as directly linked (see Fig. 2).

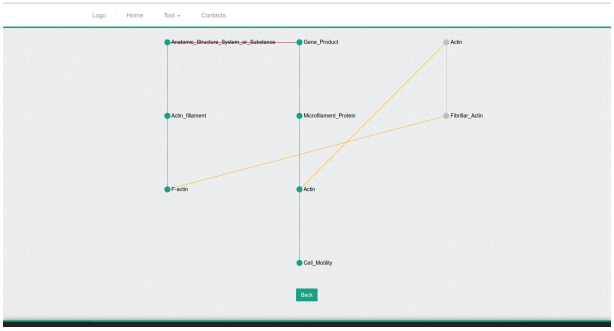


Fig. 2. Our webtool displaying the conflict set described in Figure 1. Green: NCIT; Gray: FMA; Red: disjoint axiom; Yellow: mappings.

4 CONCLUSION

Given the error rates of current repair algorithms (Pesquita *et al.*, 2013), supporting user involvement in alignment repair is key to providing coherent and correct alignments. Understanding the impact of incoherence in ontology alignments is crucial to support an effective use of alignments, which depending on the task at hand, should enforce or relax coherence. This can be of particular importance for biomedical ontologies where it is fairly common that ontologies covering the same domain are based on incompatible models. So for instance, if the task is to support the cross-references between ontologies, then coherence can be sacrificed to achieve a greater coverage, whereas in more complex tasks that depend on reasoning, such as querying support (Solimando *et al.*, 2014), ensuring coherence is paramount.

Our webtool supports users in this task by allowing them to visualize all classes, mappings and axioms involved in a logical conflict. We are currently extending the tool to permit users to manually solve conflicts, and export the repaired alignments. In future work we would like to link our tool to BioPortal, to support access to all ontologies and mappings it provides.

ACKNOWLEDGEMENTS

The authors are grateful to Daniel Faria for providing some of the data used in this work. This work was partially supported by FCT through funding of LaSIGE Research Unit, ref.UID/CEC/00408/2013. Ernesto Jimenez-Ruiz was supported by the EPSRC projects MaSI^3, Score! and DBOnto, and by the EU FP7 project Optique (grant agreement 318338).

REFERENCES

Euzenat, J., Shvaiko, P., *et al.* (2007). *Ontology matching*, volume 18. Springer.

Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I. F., and Couto, F. M. (2013). The Agreementmakerlight ontology matching system. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pages 527–541. Springer.

Faria, D., Jiménez-Ruiz, E., Pesquita, C., Santos, E., and Couto, F. M. (2014). Towards annotating potential incoherences in BioPortal mappings. In *The Semantic Web–ISWC 2014*, pages 17–32. Springer.

Fu, B., Noy, N. F., and Storey, M.-A. (2013). Indented tree or graph? A usability study of ontology visualization techniques in the context of class mapping evaluation. In *The Semantic Web–ISWC 2013*, pages 117–134. Springer.

Ivanova, V. and Lambrix, P. (2014). User Involvement for Large-Scale Ontology Alignment. In *International Workshop on Visualizations and User Interfaces for Knowledge Engineering and Linked Data Analytics*, pages 34–47.

Jiménez-Ruiz, E. and Grau, B. C. (2011). Logmap: Logic-based and scalable ontology matching. In *The Semantic Web–ISWC 2011*, pages 273–288. Springer.

Meilicke, C. and Stuckenschmidt, H. (2008). Incoherence as a basis for measuring the quality of ontology mappings. In *In Proceedings of the 3rd ISWC international workshop on Ontology Matching*, pages 1–12.

Pesquita, C., Faria, D., Santos, E., and Couto, F. M. (2013). To repair or not to repair: reconciling correctness and coherence in ontology reference alignments. In *Ontolpgy Matching workshop at ISWC 2013*, pages 13–24.

Pesquita, C., Faria, D., Santos, E., Neefs, J.-M., and Couto, F. M. (2014). Towards visualizing the alignment of large biomedical ontologies. In *Data Integration in the Life Sciences*, pages 104–111. Springer.

Santos, E., Faria, D., Pesquita, C., and Couto, F. (2013). Ontology alignment repair through modularization and confidence-based heuristics. *arXiv preprint arXiv:1307.5322*.

Solimando, A., Jiménez-Ruiz, E., and Pinkel, C. (2014). Evaluating ontology alignment systems in query answering tasks. In *ISWC 2014 Posters and Demonstrations Track*.

Whetzel, P. L., Noy, N. F., Shah, N. H., Alexander, P. R., Nyulas, C., Tudorache, T., and 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**(suppl 2), W541–W545.

Ontology-driven patient history questionnaires

Jonathan Bona[1], Gunther Kohn[2] and Alan Ruttenberg[1]
[1] Department of Oral Diagnostic Sciences, University at Buffalo, 355 Squire Hall, Buffalo NY, USA
[2] School of Dental Medicine Office of Information Resources, University at Buffalo, 108 Squire Hall, Buffalo NY, USA

1 INTRODUCTION

We are developing an ontology-driven system for collecting, recording, and managing patient histories. It consists of a web application with an RDF triple store database populated with representations in OWL of the entities that medical histories are about. It also represents the processes involved in collecting patient histories, including the questions, answers, and other information artifacts involved. It replaces paper questionnaires previously used for both general health and oral health history. The underlying ontology will be made publicly available. This abstract discusses the system, focusing on the ontological models underlying it.

2 PATIENT HISTORY COLLECTION

The UB dental school conducts education and research in addition to providing patient care. Our patient history system collects and stores data in a way that preserves its meaning independent of any one use. Its representations of the patient's history are independent of how that information was produced (i.e., the particular software used). Storing OWL representations in a triple store with reasoning makes the data readily available for queries and logical inference.

Rather than recreating paper forms in software as a list of questions, our system dynamically adjusts which questions are shown based on preceding answers. It captures provenance of any data that it records and explicitly represents the question-taking process, its participants, its sub-processes, and its results. Every answer is linked to the person who recorded it, the time, the patient it's about, etc.

The data model uses science-based ontologies associated with the OBO Foundry including Ontology of Biomedical Investigations (OBI), Information Artifact Ontology (IAO), Ontology for General Medical Science (OGMS), and Oral Health and Disease Ontology (OHD) (Scheuermann et al, 2009)(Brinkman et al, 2010)(Schleyer et al, 2013).

The system facilitates collecting general health history, oral history, family history, etc. When a patient visits our clinic for the first time, that encounter includes creating a record of the patient's history. A provider asks the patient questions and records the answers in the system. When a student in the provider role finishes history-taking, a faculty member reviews and approves the student's work. Some non-student providers can approve their own work, but all entries pass through the *unapproved* state and require approval before they become part of the patient's record.

Updating a history is similar: a provider reviews it with the patient and makes any necessary changes. The system retains the entire history of changes to the record.

3 UNDERLYING MODEL

The model underlying this system contains representations of information artifacts such as questionnaires and their contents (questions, acceptable answers); specifications governing how and when these things are to be displayed during use of the system; and workflows realized in the system. It also contains representations of entities and processes relevant to the patient's health (the patient's body, its disorders), and information about healthcare processes (a history-taking, the encounter it is a part of).

3.1 Representing the questionnaire and its parts

Figure 1 shows our representation scheme for the questions, potential answers, and groupings of those elements that comprise a questionnaire. Here we focus on a single question, though a questionnaire typically includes multiple groups of questions, multiple questions per group, and specifications for ordering those.

The questionnaire (**form1**) instantiates *IAO: document*. It has as parts *question group specification*s (**question-group1**, e.g.), which have as parts *question specification*s. A *question specification* (e.g. **question1**, about myocardial infarction) is an *IAO: directive information entity* that includes the text of the question and has as part an *answer group specification* with acceptable answers. There are a many *answer group specification* types. **answer-group-1** is an instance of the simplest: a list of labeled possible answers, one of which is to be selected as the answer. Each is an *answer specification*, (**answer-spec3**, et al).

3.2 Representing question answering

Figure 2 shows our representation of answers, the processes that produce them, and their participants.

The patient (**patient1**) has the role *patient role* throughout the visit. Each provider the patient interacts with during the visit realizes the *provider* role during that interaction. The patient's visit to the dental clinic (**encounter1**) is an *OGMS: health care encounter* that usually has other encounters as parts. The *OGMS: clinical history taking* in which **patient1**

and **provider1** participate (**history-taking1**) is one of the parts of **encounter1**. **history-taking1** has processual parts that are instances of *history question taking*. Each has a *question specification* (e.g. **question1** from Figure 1, shown here without its text) as input and produces as output a *medical history answer* (**answer1**).

Every instance of *medical history answer* is created as the output of a *history question taking*. Answer instances are connected to, but distinct from, instances of *answer specification*. There is only one answer specification for the short text answer "yes" (i.e. **answer-spec3**) but many *medical history answer*s will use/*denote* it (e.g. **answer1**). **answer1** is about a myocardial infarction that inhered in the patient at

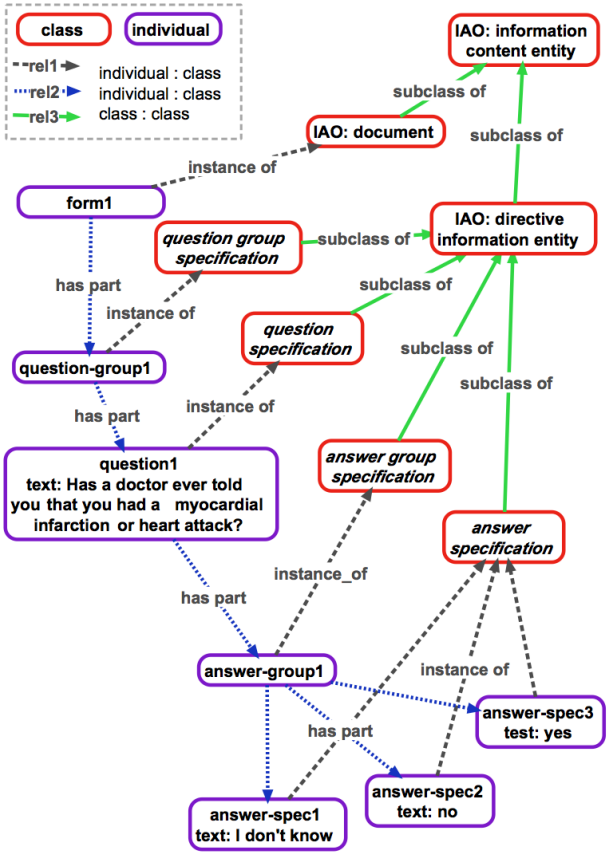


Figure 1: Specifying history questionnaire components

some time. When this question is answered "no" rather than "yes," there's no individual disorder that the answer's about.

This discussion ignores temporal considerations for simplicity's sake. Note that every answer is a unique instance and the output of a unique process of asking the question. The answer persists even when the world and knowledge about it changes, for example because a patient who had never had a myocardial infarction as of their first visit *does* experience one between their first and second visits to the clinic. When the patient's history is updated to reflect this, the old answer remains part of the record and a brand new

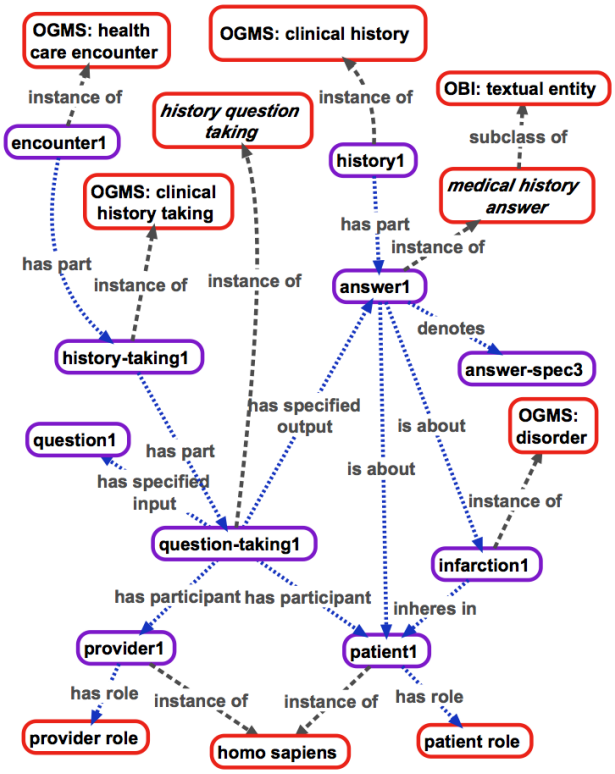


Figure 2: Question answering

instance of *medical history answer* is created as the output of a new *medical history taking* that involves the same patient and the same question.

4 CONCLUSION

We are developing a medical history application based on a carefully constructed ontological model that represents not only the things that a patient history is about, but also elements of the history-taking process, including the questionnaire and its contents. The result is a flexible, easily-queried knowledge base of patient histories with semantic representations that facilitate its use for research and in conjunction with other information systems. This work is ongoing. We continue to develop the software and representation.

REFERENCES

Brinkman RR, Courtot M, Derom D, Fostel JM, He Y, Lord P, Malone J, Parkinson H, Peters B, Rocca-Serra P, Ruttenberg A, Sansone S, Soldatova LN, Stoeckert CJ, Turner JA, Zheng J (2010) *Modeling biomedical experimental processes with OBI. J Biomed Semantics* 2010

Scheuermann, R.H., Ceusters, W. and Smith, B. (2009) *Toward an Ontological Treatment of Disease and Diagnosis. Proc. of AMIA 2009 Summit on Translational Bioinformatics*, 116-120.

Schleyer TK, Ruttenberg A, Duncan W, et al. (2013) *An ontology-based method for secondary use of electronic dental record data. AMIA Jt Summits Transl Sci Proc* 2013 234–8

Part III

Poster Abstracts

Development of a discharge ontology to support postanesthesia discharge decision making

Lucy L Wang*, Yong Choi

Department of Biomedical Informatics and Medical Education, School of Medicine, University of Washington, 850 Republican St, Seattle, WA 98195, USA

ABSTRACT

Postanesthesia discharge decision making is a challenging process due to the high complexity and variability of care provided to postoperative patients. We built an ontology-based decision support system that generates discharge recommendations for patients who have undergone surgical procedures. Discharge decisions are made based on patient vitals, symptoms, medical history and details of the surgical procedure. The output recommendations of our system can aid healthcare providers in discharge decision-making and potentially reduce readmissions due to improper discharge. This project demonstrates the potential uses of ontologies in medical decision support systems, especially in areas that use specific scoring guidelines to aid decision-making.

1 INTRODUCTION

Evidence-based discharge decision making and planning is a critical care process that can improve patient outcomes and reduce readmission rates. Inappropriate discharge can cause additional pain and suffering for patients and their families and consume unnecessary hospital resources (Anderson *et al.*, 2011). For surgical procedures, the risks associated with early discharge may be even higher. When planning for discharge, healthcare professionals have to account for multiple variables such as age, vitals, comorbidities, medications and social issues. A tool like the Aldrete scoring system is commonly used to help healthcare professionals determine when patients can be safely discharged (Aldrete, 1995). However, there are no standardized guidelines routinely used by healthcare professionals to assist them in making postoperative discharge decisions. A knowledge-based decision support tool based on standardized procedures can enhance discharge decision making and reduce errors. In Bouamrane *et al.*, 2010, the authors built an ontology to model preoperative domain knowledge. In this paper, we use a similar approach to create a postoperative ontology-based decision support system to assist discharge decision making.

2 METHODS

Our goal is to create an ontology to aid in post-surgery discharge decision-making. Following surgery, patients generally go from phase I postanesthesia care to phase II before being discharged to home. Phase I care immediately follows surgery and involves intensive monitoring of patient status. Phase II care is less intensive and sees the patient recovering well from anesthesia. The goals of our system are (i) to detect patients who may be suitable for discharge, (ii) to determine the appropriate discharge workflow, and (iii) to generate a list of additional recommendations for physicians.

*To whom correspondence should be addressed: lucylw@uw.edu

Many clinics have specified their own modified criteria for postanesthesia discharge. We begin by assembling resources published online by various surgical units. Among these resources, many are based on the Aldrete scoring system, with additional modifications tailored to clinic-specific workflow. Criteria from Stanford Hospital and Clinics, Loyola University Medical Center and others are used to construct a global discharge rule set (Stanford Hospital and Clinics, 2010; Brown *et al.*, 2008). The Phillips *et al.*, 2011 systematic review of postanesthesia discharge protocols is also used to determine levels of evidence for various scoring criteria. Scoring guidelines present in all or most resources we studied are included as criteria in our ontology-based decision support system.

Based on the the criteria in these resources, we build a set of SWRL rules, which in turn guides our development of an OWL ontology. We first create a full set of postanesthesia discharge criteria using information from our source documents. We then translate these criteria into SWRL rule syntax to facilitate reasoning. Afterwards, we use these rules to guide the creation of OWL classes, as well as the definition of object and data properties.

The modified Aldrete score, for example, consists of five primary criteria: consciousness, respiration, circulation, movement and pain. Some of these, such as respiration and circulation, can be broken down further. For example, respiration consists of breathing quality, breath rate and oxygen saturation.

Aldrete subscores, along with the total score, are used as criteria for discharge. For each Aldrete subscore, a patient receives a score on a 2 point scale, where 0 means a low functional level and 2 means a normal functional level. We create a data property that corresponds to each primary criteria. The value of this data property is determined using SWRL rules and assigned to a patient based on his/her current status in the system. An example data property *hasAldreteScoreConsciousness* may take on the values of 0, 1 or 2 if the patient is unresponsive, responsive but drowsy, and responsive and fully alert respectively.

Another example is the circulation subscore, where the patient's blood pressure must fall within a pre-specified range from the baseline blood pressure. Within $\pm$ 20 mmHg yields a score of 2, within $\pm$ 20-50 mmHg yields a score of 1, and anything outside of that range yields a score of 0. These specific differences can be automatically calculated by our reasoner, which then assigns a score to the patient. This reduces the need for healthcare workers to perform time-consuming numerical calculations.

An example SWRL rule for oxygen saturation is:

$$Patient(?pt), hasSpO2(?pt, ?SpO2), greaterThan(?SpO2, 95) \rightarrow hasAldreteScoreOxygen(?pt, 2)$$

which assigns 2 points to the Aldrete oxygen subscore if a patient's oxygen saturation is greater than 95%. The sum of points

assigned to all Aldrete criteria is then calculated and used to determine whether a patient fits the basic criteria for discharge. If a patient satisfies this condition, she/he is assigned into the class *DISCHARGE_FROM_PHASE_I_POSTANESTHETIC_CARE*.

Our ontology classifies patients for discharge from phase I to phase II care, as well as from phase II care to home. Additionally, our discharge ontology makes recommendations for healthcare provider actions. For example, a patient receiving a sciatic block may require clutches at discharge (Figure 1), or a patient with high pain levels may require additional pain management. These recommendations can be used by healthcare providers to prioritize patient care and to generate discharge notes.

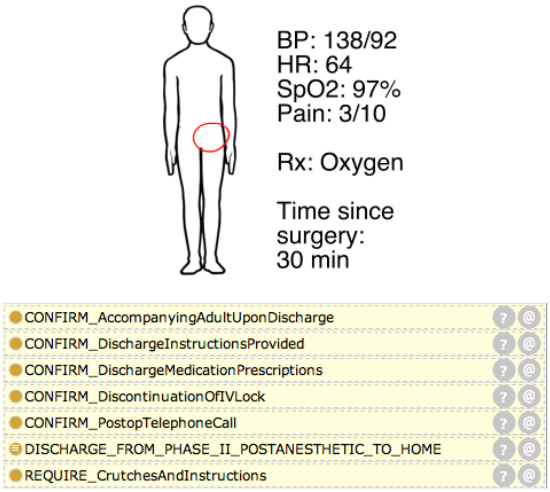


Fig. 1. Example patient who is 30 minutes post surgery with stable vitals. Output classifications based on SWRL rules are given.

3 RESULTS & DISCUSSION

In this project, we demonstrated our work in building a knowledge-based decision support system that generates decision support recommendations to determine patient discharge eligibility after surgical procedures. We were able to model appropriate discharge decision making in several example patients (Figure 1). In addition to making the correct discharge decision, our system also generates a list of recommendations for clinicians which should be followed before the actual discharge.

Our decision support system operated with a number of limitations. First of all, the recommendations and guidelines issued by our system are constrained by the accuracy of the guidelines that we modeled. Therefore, any errors or flaws present in the model guidelines will be systematically replicated by our system. Also, due to the lack of a standard discharge protocol, we could only capture a representative set of criteria. Our ontology, therefore, may need to be modified for use in any specific clinical environment.

Additionally, we should align our system with pre-existing medical ontologies for morbidity classification such as ICD-10 or SNOMED-CT. We believe that such integration is critical for the future interoperability of our system. Future work will also involve extracting information from electronic health records or in-room patient sensors (Figure 2) to increase the accuracy, timeliness and reliability of patient medical data. Some obvious challenges to this work are the semantic plurality of clinical data representation and non-standard data exchange protocols between platforms.

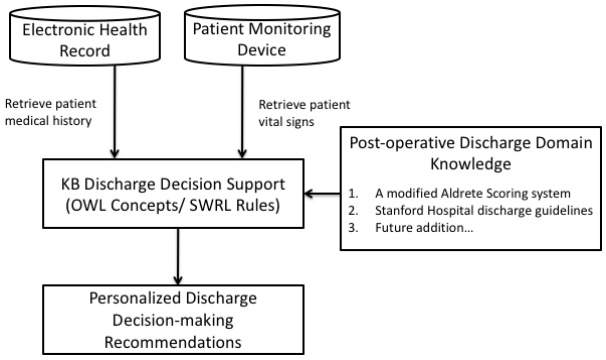


Fig. 2. Integrating information from electronic health records and patient monitoring devices into the decision support ontology.

Another future direction is to expand our system to create automated discharge summary notes to assist in the transition of care. The discharge summary notes can be generated for two groups of users: (1) healthcare professionals, and (2) patients and caregivers. Discharge notes generated for healthcare professionals can be used to facilitate continuity of care. Notes generated for patients and caregivers can contain care instructions specifically tailored to the patient to help guide them through the complex post-discharge care process.

4 CONCLUSION

The discharge decision-making process relies on a set of predefined clinical criteria that must be interpreted correctly to reach the appropriate discharge decision. Our ontology integrates patient vitals, symptoms, and surgical and medical information, and outputs recommendations for discharge and healthcare provider actions. This decision support tool could simplify postanesthesia discharge procedures and may help reduce adverse events based on improper or early discharge.

ACKNOWLEDGEMENTS

This study was supported in part by National Library of Medicine (NLM) Training Grant T15LM007442.

REFERENCES

Aldrete, J. A. (1995). The post-anesthesia recovery score revisited. *J. Clin. Anesth.*, **7**, 89–91.

Anderson, D., Price, C., Golden, B., Jank, W., and Wasil, E. (2011). Examining the discharge practices of surgeons at a large medical center. *Health Care Manag. Sci.*, **14**, 338–347.

Bouamrane, M. M., Rector, A., and Hurrell., M. (2010). Experience of using owl ontologies for automated inference of routine pre-operative screening tests. *International Semantic Web Conference 2010*, pages 50–65.

Brown, I., Jellish, W. S., Kleinman, B., Fluder, E., Sawicki, K., Katsaros, J., and Rahman, R. (2008). Use of postanesthesia discharge criteria to reduce discharge delays for inpatients in the postanesthesia care unit. *J. Clin. Anesth.*, **20**, 175–179.

Phillips, N. M., Haesler, E., Street, M., and Kent, B. (2011). Post-anaesthetic discharge scoring criteria: a systematic review. *JBI Library of Systematic Reviews*, **9**, 1679–1713.

Stanford Hospital and Clinics (2010). Discharge criteria for phase I & II post anesthesia care.

Improvements to the Drosophila anatomy ontology

Marta Costa*, David Osumi-Sutherland, Steven Marygold and Nick Brown
FlyBase, Department of Genetics, University of Cambridge, Downing Street, Cambridge, UK

ABSTRACT

The *Drosophila* anatomy ontology (DAO) defines the broad anatomy of the fruitfly *Drosophila melanogaster*, a genetic model organism. It contains over 8700 classes, with close to half of these corresponding to neuroanatomical terms.

We are systematically reviewing the DAO classes, improving the textual information and classification. This includes adding definitions, comments and synonyms, as well as formal definitions, which results in a full classification in some cases. Classes belonging to each of the defined organ systems are reviewed together to improve consistency of free text and formalisation. So far we have reviewed 7 of the 11 organs system classes, resulting in 83% of classes having a definition.

1 INTRODUCTION

The *Drosophila* anatomy ontology (DAO) (Costa *et al.*, 2013) is an ontology that describes the wild-type anatomy of *Drosophila*, containing over 8700 classes. It is used by FlyBase (Dos Santos *et al.*, 2015), the gene and genomic database for *Drosophila*, for manual curation of phenotypes and expression patterns. Users are also able to query for this type of data, either through FlyBase or Virtual Fly Brain (Milyaev *et al.*, 2012). Having an accurate, encompassing and human-readable ontology is therefore essential to enable curators to choose the correct anatomy term, and for users to easily navigate the data.

When the DAO was first developed over 20 years ago, it did not include textual information or significant formalisation. A large effort has been undertaken in the last 9 years to improve this situation (Costa *et al.*, 2013). This work has resulted in 83% of classes now having a definition and the classification having been greatly improved. The DAO currently contains 46 object properties and over 18,000 subclass axioms, with over 2,500 equivalent class axioms, with around 50% of over 10,000 classifications being inferred.

New DAO classes are curated from the published literature, if enough evidence regarding their morphological characterisation, identity and if appropriate, function, are provided. Every class includes a definition, synonyms and comments each attributed to a source reference.

The neuroanatomy field has grown massively in the last few years thanks to technical advances, enabling researchers to identify and characterize the function of individual neurons and several projects are currently underway to map all neurons in a variety of model organisms. We, in collaboration with the Virtual Fly Brain project, have focused on capturing this information in *Drosophila*. Currently, 46% of the DAO comprises terms that are part of the nervous system, including close to 2300 distinct neuron classes, over 220 neuroblast lineage clones and neuropils.

Here, we present our most recent work in improving the textual information and formalisation patterns of DAO classes.

2 RESULTS

In order to maintain consistency between related terms, we are reviewing existing classes by making use of their current classification into 11 different organ systems, such as the tracheal, muscle, adipose, etc. Around 80% of classes in the DAO had previously been classified as part of an organ system previously, thus making it easy to retrieve a list of classes to review. Work has proceeded class by class, improving both the textual information and formalisation. When necessary, we have sought advice from expert researchers.

The systematic review of classes uncovered several cases of redundancy and duplication, which were resolved by obsoleting one of the terms. For example, the classes Malpighian tubule Type II cell (see section 2.1) and excretory star cell were found to refer to same entity. In this case, the latter was obsoleted, and the name added as a synonym to the former.

We have concluded this review for 7 of the 11 organ systems (muscle, tracheal, reproductive, digestive, circulatory, excretory and adipose), corresponding to 570 classes. Work is ongoing to complete the remaining (muscle, nervous, endocrine and sensory).

2.1 Improving textual information

We have added textual definitions to 83% of DAO classes, an improvement of 10% since October 2013. The definition describes the general classification of the anatomical entity, its properties, and when appropriate, any distinguishing traits. These statements are supported by references, either cited in the text, or listed at the end with a publication identifier (mostly a FlyBase one: prefix FBrf followed by 7 dig-

* To whom correspondence should be addressed: m.costa@gen.cam.ac.uk

its).

Comments are added when appropriate, for one of two reasons. The first is to provide relevant information relating to the experimental setup when, for example, investigating the function of a neuron. The second is to clarify the relationship between competing nomenclatures.

Synonyms from the published literature are added to each class, together with references. The addition of synonyms has particular relevance to anatomy ontologies, for which competing nomenclatures often exist.

An example of the textual information for a class is below:

> **name:** Malpighian tubule Type II cell
>
> **definition:** "Morphologically distinct cell type found only in the initial, transitional and main segments of the Malpighian tubules interspersed with Type I cells. Type II cells are smaller and flatter than Type I cells, with shorter (main segment) or no (initial region) apical microvilli. Type II cells originate from a subset of caudal visceral mesoderm cells that overlie the tubule primordia as they evert from the hindgut. By stage 15, Type II cells have been incorporated in the tubules and adopt epithelial characteristics. In the mature tubules there are on average 110 Type II cells." [FlyBase:FBrf0064792, FlyBase:FBrf0102373, FlyBase:FBrf0160477, FlyBase:FBrf0222532]
>
> **comment:** These cells are involved in primary urine production via the presence of ion channels that allow chloride and water to enter the tubule lumen (O'Donnell et al., 1998).
>
> **synonyms:** "excretory star cell" EXACT; "Malpighian tubule stellate cell" EXACT [FlyBase:FBrf0030988]

2.2 Improving formal definitions

In systems such as the tracheal, in which certain structures are repeated in each metameric unit, adding a formal definition to each of these terms significantly increases the robustness of error checking procedures. An example of some of the relationships that are added is below:

> **name:** adult abdominal spiracular branch
>
> **intersection_of:** FBbt:00003071 ! adult spiracular branch
>
> **intersection_of:** connected_to FBbt:00003040 ! adult lateral trunk
>
> **intersection_of:** connected_to FBbt:00004814 ! adult abdominal spiracle
>
> **intersection_of:** part_of FBbt:00003024 ! adult abdominal segment

In other instances, adding a formal definition allows for full classification of terms. This becomes particularly relevant for neuroanatomy, a field in which new neuron types are being frequently described. Having a formal definition for a class ensures that new terms are correctly classified, provided that enough information is available, such as developmental origin.
An example of a neuron class that can be fully classified based on expression (which identifies this subset of very well studied neurons) and developmental origin is below. This formalisation pattern, or a similar one (excluding only the neuroblast information), was used to define the 104 classes of adult *fruitless* neurons.

> **name:** adult fruitless aDT-b (female) neuron
>
> **intersection_of:** FBbt:00005106 ! neuron
>
> **intersection_of:** develops_from FBbt:00050148 ! neuroblast CREa1 (female)
>
> **intersection_of:** expresses FlyBase:FBgn0004652 ! fruitless
>
> **intersection_of:** part_of FBbt:00110416 ! adult fruitless aDT-b (female) lineage clone

3 DISCUSSION

We have reviewed terms that belong to 7 of the 11 organ systems in the DAO, improving the textual information essential for casual users, and the formalisation necessary to easily maintain a correct classification and to prevent the introduction of errors. Reviewing related classes as a group helps to maintain consistency in the ontology, both in terms of free text and the formalisation patterns used.
Future work will focus on completing the systematic review of the DAO by revising the classes in the remaining 4 organ systems.

REFERENCES

Costa, M., Reeve, S., grumbling, G. and Osumi-Sutherland, D. (2013). The Drosophila anatomy ontology. *J. Biomedical Semantics*, 32-4.

Dos Santos G, Schroeder AJ, Goodman JL, Strelets VB, Crosby MA, Thurmond J, Emmert DB, Gelbart WM; the FlyBase Consortium. (2015). FlyBase: introduction of the Drosophila melanogaster Release 6 reference genome assembly and large-scale migration of genome annotations.
Nucleic Acids Res. doi: 10.1093/nar/gku1099

Milyaev, N., Osumi-Sutherland, D., Reeve, S., Burton, N., Baldock, R. A. and Armstrong, J. D. (2012). The Virtual Fly Brain browser and query interface. *Bioinformatics* **28**, 411-5.

Onto-animal tools for reusing ontologies, generating and editing ontology terms, and dereferencing ontology terms

Yongqun He[1*], Jie Zheng[2], Yu Lin[1]
[1] University of Michigan, Ann Arbor, Michigan, USA; [2] University of Pennsylvania, Philadelphia, PA, USA

ABSTRACT

Onto-animal tools are a package of web-based ontology tools developed to support efficient and integrated ontology development and application. This package of tools includes OntoFox, Ontodog, Ontorat, Ontobee, Ontobeep, and Ontobat. Each tool has specific functions; together, these tools support the extraction of a single or subset of terms and community views from existing ontologies, generation and editing of ontology terms, query and visualization of ontology terms, comparison among ontologies, and instance-level data representation and analysis. Based on the Web Ontology Language (OWL) and Semantics Web technologies, these tools have widely been used by thousands of ontology developers in over 20 communities.

1 INTRODUCTION

Biological/biomedical ontologies are sets of computer- and human-interpretable terms and relations that represent entities and their relations in the biological/biomedical world. Biomedical ontologies have emerged as a major tool for the integration and analysis of the large amounts of heterogeneous biological data available in the post-genomics era.

To support ontology development and applications, we have developed a collection of "Onto-animal" tools, including OntoFox (Xiang et al., 2010), Ontodog (Zheng et al., 2014), Ontorat (Xiang et al., 2015), Ontobee (Xiang et al., 2011), Ontobeep (Xiang and He, 2010), and Ontobat (Xiang et al., 2015). The back-end of these Onto-animal tools is the He group's RDF triple store (http://sparql.hegroup.org), which has become the default ontology RDF triple store for the Open Biological and Biomedical Ontologies (OBO) Foundry ontologies. Fig. 1 provides a summary of these tools.

Although initially developed to meet the needs of Vaccine Ontology (VO) development (He et al., 2009;Ozgur et al., 2011), the "Onto-animal" tools have been widely used in a broader range of users for various applications. According to Google Analytics, in the past five years, over 8,000 and 35,000 users from >10 countries have routinely used the OntoFox and Ontobee web programs, respectively. According to Google Scholar, our Onto-animal tools have been cited in approximately 200 publications.

To provide a whole picture of how these tools work and interact, here we briefly introduce general features of each Onto-animal tool and how these tools can be used to support different applications.

2 ONTO-ANIMAL TOOLS

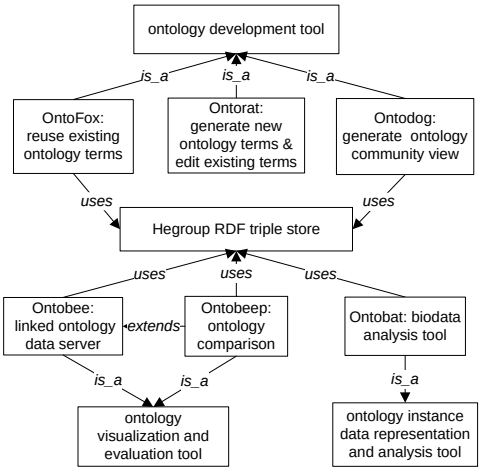


Fig. 1. Onto-animal tools and their features

2.1 OntoFox: Extract ontology terms and axioms

Reusing a portion of an existing ontology is often required in the ontology development process. After a user provides a single or a set of terms of interest, OntoFox (http://ontofox.hegroup.org/) is able to fetch selected classes, properties, annotations, and their related terms from source ontologies and save the results in the OWL format (Xiang et al., 2010). OntoFox implements the Minimum Information to Reference an External Ontology Term (MIREOT) strategy by extracting minimum information of requested terms (Courtot et al., 2011). In addition, by providing different options, OntoFox can extract different levels of intermediate terms between the required terms and a chosen higher level or top term. Inspired by existing ontology modularization techniques (Stuckenschmidt et al., 2009). OntoFox also implements a new SPARQL-based ontology term extraction algorithm that extracts all terms and axioms related to a given set of user-provided terms (Xiang et al., 2010).

2.2 Ontodog: Generate ontology community view

Similar to OntoFox, the web-based Ontodog program (http://ontodog.hegroup.org/) is able to extract a subset of ontology terms and axioms (Zheng et al., 2014). Unlike OntoFox, Ontodog includes two unique features. First, Ontodog allows the generation of an ontology community view, which we have defined as "the whole or a subset of the source ontology with *user-specified annotations includ-*

* To whom correspondence should be addressed:yongqunh@umich.edu

ing user preferred labels" (Zheng et al., 2014). Second, Ontodog uses Excel input files to identify which terms to retrieve and to add user-specified annotations. Excel templates are also provided for easy implementation.

2.3 Ontorat: Adding new terms and new axioms to an ontology based on design pattern

The Ontorat program (http://ontorat.hegroup.org) automatically generates and edits ontology terms and axioms and provides term annotations (Xiang et al., 2015). Ontorat uses reusable ontology design patterns (ODPs) to solve recurrent modeling problems. A specific ODP can be used to derive an Excel template of different terms/annotations and a set of rules that define the relations among those terms/annotations. An Ontorat template is similar to a QTT (Quick Term Template) (Rocca-Serra et al., 2011). Such a template can be populated with specific terms or annotations to define or annotate specific ontology terms. With the support of the Ontorat settings, the populated template spreadsheet can then be converted into an OWL file with newly generated ontology terms and axioms.

2.4 Ontobee: Linked data server for web displaying and dereferencing ontology terms

Ontobee (http://www.ontobee.org) is an ontology browser and a linked ontology data server for dereferencing ontology terms (Xiang et al., 2011). To date, there are 156 ontologies listed on Ontobee. Ontobee loads individual page for each term in an ontology. All related information of a single term, such as label, definition, synonyms, superclass hierarchy, logical axioms, and term usage by other ontologies is displayed. In addition, for each ontology, Ontobee generates statistics with counts of classes, object properties, annotation properties, and datatype properties based on term ontology prefixes. Furthermore, Ontobee automatically provides an Excel document listing all terms in an ontology. Ontobee provides ontology term search and SPARQL query service supported by the He group triple store. Ontobee is a *de facto* search engine for OBO Foundry ontologies.

2.5 Ontobeep: Ontology comparison

Ontobeep (http://www.ontobee.org/ontobeep/) is an ontology comparison program. Ontobeep can be used to compare different ontologies by aligning them from the roots of these ontologies. The alignment identifies common terms existing in two or three ontologies. Ontobeep also provides a statistic report of the alignment analysis. Ontobeep may be utilized to detect inconsistency and term duplication in one or more ontologies.

2.6 Ontobat: Ontology-based data analysis

Unlike other Onto-animal tools, Ontobat (http://ontobat.hegroup.org) focuses on instance level ontology data generation and analysis (Xiang et al., 2014). Ontobat aims to support Linked Open Data (LOD) generation, upload, query, browsing, and statistical analysis. Many features of Ontobat are still under development. .

3 SUMMARY

The web-based Onto-animal tool package provides a set of comprehensive tools to support ontology development and applications. These tools save time and efforts for ontology developers and users, especially those who do not have or have limited software programming background.

ACKNOWLEDGEMENTS

This research was supported by a NIH R01 grant (1R01AI081062).

REFERENCES

Courtot, M., Gibson, F., Lister, A., Malone, J., Schober, D., Brinkman, R., and Ruttenberg, A. (2011). MIREOT: the Minimum Information to Reference an External Ontology Term. *Applied Ontology* 6**,** 23-33.

He, Y., Cowell, L., Diehl, A.D., Mobley, H.L., Peters, B., Ruttenberg, A., Scheuermann, R.H., Brinkman, R.R., Courtot, M., Mungall, C., Xiang, Z., Chen, F., Todd, T., Colby, L.A., Rush, H., Whetzel, T., Musen, M.A., Athey, B.D., Omenn, G.S., and Smith, B. (Year). "VO: Vaccine Ontology", in: *The 1st International Conference on Biomedical Ontology (ICBO-2009)*: Nature Precedings), http://precedings.nature.com/documents/3552/version/3551.

Ozgur, A., Xiang, Z., Radev, D.R., and He, Y. (2011). Mining of vaccine-associated IFN-gamma gene interaction networks using the Vaccine Ontology. *J Biomed Semantics* 2 Suppl 2**,** S8.

Rocca-Serra, P., Ruttenberg, A., O'connor, M.J., Whetzel, P.L., Schober, D., Greenbaum, J., Courtot, M., R.R., B., S.A., S., R., S., Consortium, T.O., and Peters, B. (2011). Overcoming the ontology enrichment bottleneck with quick term tem-plates. *Applied Ontology* 6**,** 13-22.

Stuckenschmidt, H., Parent, C., and Spaccapietra, S. (2009). *Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization.* Springer.

Xiang, Z., Courtot, M., Brinkman, R.R., Ruttenberg, A., and He, Y. (2010). OntoFox: web-based support for ontology reuse. *BMC Res Notes* 3:175**,** 1-12.

Xiang, Z., and He, Y. (2010). IDO extensions alignment using Ontobeep. IDO Workshop 2010. URL: http://ontology.buffalo.edu/2010/IDO/xiang.pptx.

Xiang, Z., Lin, Y., and He, Y. (Year). "Ontobat: An Ontology-based semantic web approach for linked data processing and analysis", in: *Proceedings of the 5nd International Conference on Biomedical Ontologies (ICBO)*: CEUR Workshop Proceedings), Pages 93-95 [http://ceur-ws.org/Vol-1327/icbo2014_paper_1358.pdf]

Xiang, Z., Mungall, C., Ruttenberg, A., and He, Y. (Year). "Ontobee: A linked data server and browser for ontology terms", in: *The 2nd International Conference on Biomedical Ontologies (ICBO)*: CEUR Workshop Proceedings), Pages 279-281 [http://ceur-ws.org/Vol-833/paper248.pdf].

Xiang, Z., Zheng, J., Lin, Y., and He, Y. (2015). Ontorat: Automatic generation of new ontology terms, an-notations, and axioms based on ontology design patterns. *Journal of Biomedical Semantics* 6**,** 4 (10 pages).

Zheng, J., Xiang, Z., Stoeckert, C.J., Jr., and He, Y. (2014). Ontodog: a web-based ontology community view generation tool. *Bioinformatics* 30**,** 1340-1342.

Bridging Vaccine Ontology and NCIt vaccine domain for cancer vaccine data integration and analysis

Yongqun He[1], Guoqian Jiang[2]
[1] University of Michigan Medical School, Ann Arbor, MI 48109, USA;
[2] Mayo Clinic, Rochester, MN, 55906, USA

ABSTRACT

The Vaccine Ontology (VO) is a community-based ontology in the domain of vaccines and vaccination. VO is aligned with the Basic Formal Ontology (BFO) and developed by following OBO Foundry principles. National Cancer Institute (NCI) Thesaurus (NCIt) serves as a reference ontology to facilitate interoperability and data sharing for cancer translational and basic research. To facilitate better cancer vaccine research, we compared the VO and NCIt vaccine domain (NCIt-vaccine) and examined the possibility of bridging and integrating these two ontologies. Our results showed that only a small portion of vaccine terms overlap between the two ontologies, and VO and NCIt-vaccine are complementary in different aspects. It is possible to integrate, map, and merge them. This study can be used as a use case for achieving the broader goal of merging and integrating NCIt and OBO library ontologies.

1 INTRODUCTION

Cancer clinical and biology research studies have generated large volumes of data. Barriers to data normalization, standardization, and quality assurance make it difficult to annotate and integrate cancer data in meaningful ways and hence delay widespread research data reuse within the broader scientific community. In cancer vaccine study domain, for example, there is an urgent need to develop an integrated data and knowledge repository that can facilitate translational research studies in developing treatment vaccines against many types of cancer. To this end, ontology-based data integration approaches have been increasingly used to address this challenge (Mate et al., 2015).

Notably, NCI has developed NCIt that serves as a reference ontology to facilitate interoperability and data sharing for cancer translational and basic research (de Coronado et al., 2004). NCI has been exploring new approaches to broaden external participation in the ontology development and quality assurance process, including introducing a solid upper-level ontology. NCIt includes a vaccine branch (NCIt-vaccine) that covers many different cancer-related vaccines. Concurrently, the Open Biological and Biomedical Ontologies (OBO) Foundry, as a collaborative initiative, has aimed at establishing a set of ontology development principles and incorporating ontologies following these principles in an evolving non-redundant and interoperable suite (Smith et al., 2007). The OBO library currently includes >160 ontologies covering >3 million terms. The OBO ontologies related to clinical and biological vaccine studies include the Vaccine Ontology (VO) (He et al., 2009;Ozgur et al., 2011).

The importance of merging and integrating NCIt and OBO library ontologies has been well recognized (de Coronado et al., 2007). Here we compared VO and NCIt-vaccine with the aim to possibly align and merge these two ontologies together. Our results show both promise and challenges.

* To whom correspondence should be addressed: yongqunh@umich.edu and Jiang.Guoqian@mayo.edu

2 METHODS

2.1. Vaccine module extraction and ontology metrics comparison

The current versions (as of April 24, 2015) of VO and NCIt (version 15.03e) were obtained from their download websites. We used an OWL-based ontology module extraction tool (https://sites.google.com/site/ontologymodularity/) and extracted the vaccine module from each respectively, anchored by the VO code "vaccine (VO_0000001)" and the NCIt code "Vaccine (C923)" and their subclasses. We compared the ontology metrics of the two vaccine modules using the Protégé 5 Ontology Metrics plugin.

2.2. Ontology alignment and coverage analysis

We first manually aligned direct subclasses of the vaccine codes in two modules, and then used a UMLS-based lexical mapping tool called the Sub-Term Mapping Tools (STMT) (Lu and Browne, 2012) to retrieve the UMLS CUIs for all subclasses of the vaccine code VO_0000001 in VO. As each NCIt code has already had a corresponding UMLS CUI asserted, we produced the mappings of vaccine terms between these two ontologies. The content coverage for the vaccine terms (matched and unmatched) between the two ontologies was analyzed.

3 RESULTS

3.1 Ontology module extraction and metrics comparison

VO currently covers 4,751 terms including ~800 terms imported from other existing ontologies (http://www.ontobee.org/ontostat.php?ontology=VO). If we only count the classes under VO:vaccine (VO_0000001),

the VO vaccine branch has 28 direct subclasses and 2,140 descendants. In comparison, the NCIt-vaccine section has 11 direct subclasses and 703 descendants (Table 1).

Ontology Metrics	VO Vaccine	NCIt Vaccine
Axiom	26704	13954
Logical axiom count	8151	1570
Class count	3047	874
Class axioms		
SubClassOf axioms count	7776	1513
EquivalentClasses axioms	144	5
DisjointClasses axioms count	8	15
Object property count	82	18
DL expressivity	SROIQ	S

Table 1. Comparison of VO and NCIt-vaccine ontology metrics

3.2. VO-NCIt vaccine domain ontology alignment

Our analysis found that 10 of 11 NCIt high-level vaccine codes had exact matches with VO vaccine codes. VO has an additional 17 subclasses (e.g., 'allergy vaccine' and 'prime-boost vaccine') that do not have any NCIt match.

For the lexical mappings, in total, 280 matches were identified for VO vaccine terms with UMLS CUIs. These may serve as bridging points between VO and NCIT. NCIt is more focused on cancer vaccines. VO is more focused on infectious disease vaccines. In addition to various vaccines, VO also represents various vaccine components such as vaccine antigens, adjuvants, DNA vaccine plasmids, etc. These can be used to logically represent specific vaccines.

3.3. Bridging VO and NCIt-vaccine

NCIt-vaccine includes many cancer vaccines not included in VO. Unlike VO vaccines, these cancer vaccines are not fully represented. Therefore, it is possible to apply VO representation methods to logically represent NCIt-vaccine. For example, we modeled the NCIt-vaccine 'Alpha Fetoprotein Plasmid DNA Vaccine' (NCIt: C48373; synonym: phAFP) (Fig. 1). This vaccine consists of a plasmid DNA encoding alpha fetoprotein. After vaccination, expressed alpha fetoprotein may stimulate a cytotoxic T lymphocyte (CTL) response against tumor cells that express alpha fetoprotein, resulting in tumor cell lysis (Hanke et al., 2002).

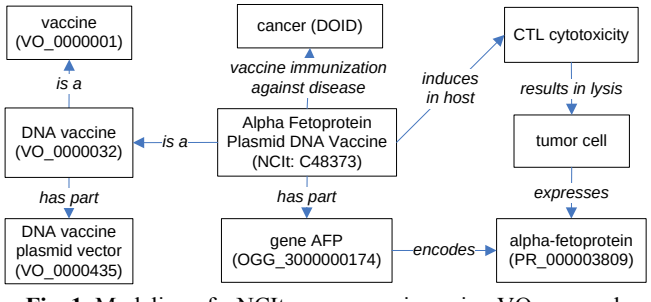


Fig. 1. Modeling of a NCIt cancer vaccine using VO approach

4 DISCUSSION

Although the topics of these ontologies are also covered by NCIt and its associated ontologies, NCIt, in general, lacks of granular terms that could be complemented by OBO ontologies that cover more terms in the biological domain. Our pilot study demonstrated that differences of ontology metrics of the vaccine modules extracted from the two ontologies, in terms of axiom richness and DL expressivity. While only a small portion of vaccine terms overlap between the two ontologies, both ontologies are complementary to each other in different ways. For better cancer vaccine study data integration, it is possible to align, map, and possibly merge these two vaccine modules. The merged ontology could be used to annotate the data and metadata available in various cancer vaccine resources, such as the CanVaxKB knowledgebase (http://www.violinet.org/canvaxkb/).

ACKNOWLEDGEMENTS

This research was supported in part by a bridge fund to Y.H. in the University of Michigan and a NCI U01 caCDE-QA grant (1U01CA180940-01A1).

REFERENCES

De Coronado, S., Haber, M.W., Sioutos, N., Tuttle, M.S., and Wright, L.W. (2004). NCI Thesaurus: using science-based terminology to integrate cancer research results. *Stud Health Technol Inform* 107**,** 33-37.

De Coronado, S., Tuttle, M.S., and Solbrig, H.R. (2007). Using the UMLS Semantic Network to validate NCI Thesaurus structure and analyze its alignment with the OBO relations ontology. *AMIA Annu Symp Proc***,** 165-170.

Hanke, P., Serwe, M., Dombrowski, F., Sauerbruch, T., and Caselmann, W.H. (2002). DNA vaccination with AFP-encoding plasmid DNA prevents growth of subcutaneous AFP-expressing tumors and does not interfere with liver regeneration in mice. *Cancer Gene Ther* 9**,** 346-355.

He, Y., Cowell, L., Diehl, A.D., Mobley, H.L., Peters, B., Ruttenberg, A., Scheuermann, R.H., Brinkman, R.R., Courtot, M., Mungall, C., Xiang, Z., Chen, F., Todd, T., Colby, L.A., Rush, H., Whetzel, T., Musen, M.A., Athey, B.D., Omenn, G.S., and Smith, B. (Year). "VO: Vaccine Ontology", in: *The 1st International Conference on Biomedical Ontology (ICBO-2009)*: Nature Precedings), http://precedings.nature.com/documents/3552/version/3551.

Lu, C.J., and Browne, A.C. (Year). "Development of Sub-Term Mapping Tools (STMT)," in: *AMIA 2012 Annual Symposium, November 3-7, 2012*), Page 1845.

Mate, S., Kopcke, F., Toddenroth, D., Martin, M., Prokosch, H.U., Burkle, T., and Ganslandt, T. (2015). Ontology-based data integration between clinical and research systems. *PLoS One* 10**,** e0116656.

Ozgur, A., Xiang, Z., Radev, D.R., and He, Y. (2011). Mining of vaccine-associated IFN-gamma gene interaction networks using the Vaccine Ontology. *J Biomed Semantics* 2 Suppl 2**,** S8.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L.J., Eilbeck, K., Ireland, A., Mungall, C.J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.A., Scheuermann, R.H., Shah, N., Whetzel, P.L., and Lewis, S. (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotechnol* 25**,** 1251-1255.

2015 Disease Ontology update: DO's expanded curation activities to connect disease-related data

Elvira Mitraka[1] and Lynn M. Schriml[1,*]
[1] Institute for Genome Science, University of Maryland School of Medicine, Baltimore, MD, USA

ABSTRACT

The Human Disease Ontology is a widely used biomedical resource, which standardizes and classifies common and rare human diseases. Its latest iteration makes use of the OWL language to facilitate easier curation between a variety of working groups and to take advantage of the analyses available using OWL. The DO integrates disease concepts from ICD-9, ICD-10, the National Cancer Institute Thesaurus, SNOMED-CT, MeSH, OMIM, EFO and Orphanet. The DO Team is focused on enabling mapping and curation of large disease datasets for major Biomedical Resource Centers and integration of their disease terms into DO. Constant updates and additions to the ontology allow for coverage of the vast field of human diseases. By having close collaborations with a variety of research groups, such as MGD, EBI, NCI, the Disease Ontology has established itself as the go-to tool for human disease curation. Implementing a combination of informatic tools and manual curation DO ensures that it maintains the highest standard possible.

1 INTRODUCTION

The Disease Ontology (DO) (Schriml *et al.*, 2012) is the core disease data resource for the biomedical community. Human disease data is a cornerstone of biomedical research for identifying drug targets, connecting genetic variations to phenotypes, understanding molecular pathways relevant to novel treatments and coupling clinical care and biomedical research. Consequently, across the multitude of biomedical resources there is a significant need for a standardized representation of human disease to map disease concepts across resources, to connect gene variation to phenotypes and drug targets and to support development of computational tools that will enable robust data analysis and integration.

2 CURRENT STATUS

DO has proven to be an invaluable genomics and genetic disease data resource used for evaluating and connecting diverse sets of data, used by diverse curation groups to connect human disease to animal models and genomic resources and used to informatically identify representative phenotype sets (Köhler *et al.,* 2014; Schofield *et al.*, 2010), functionally similar genes (Fang and Gough, 2013; Singleton *et al.*, 2014), human gene and genome annotations (Peng *et al.*, 2013; Osborne *et al.*, 2009), pathways, cancer variants (Wu *et al.,* 2014) and immune epitopes (Vita *et al.,* 2014). The DO website (http://www.disease-ontology.org), is a web-based application that allows users to query, browse, and visualize the Disease Ontology and disease concept data.

The latest version of DO has close to 9,000 disease terms, more than 16,000 synonyms and almost 39,000 cross-references to other biomedical resources. Those resources include the ICD-9 and ICD-10, the National Cancer Institute (NCI) Thesaurus (Sioutos *et al.,* 2007), SNOMED-CT (Donnelly, 2006) and MeSH (https://www.nlm.nih.gov/mesh/MBrowser.html) extracted from the Unified Medical Language System (UMLS) (Bodenreider, 2004) based on the UMLS Concept Unique Identifiers for each disease term. DO also includes disease terms extracted directly from Online Mendelian Inheritance in Man (OMIM) (Ambereger *et al.,* 2011), the Experimental Factor Ontology (EFO, http://www.ebi.ac.uk/efo/) and Orphanet (Maiella *et al.,* 2013).

The DO files are available in both OBO and OWL format from DO's SourceForge site (http://sourceforge.net/p/ diseaseontology/code/HEAD/tree/trunk) and can be found at http://purl.obolibrary.org/obo/doid.obo and http://purl.obolibrary.org/obo/doid.owl. DO's OBO and OWL files are also available from the OBO Foundry (http://www.obofoundry.org/cgi-bin/detail. cgi?id=disease ontology) and GitHub (https://github.com/obophenotype/human-disease-ontology/tree/master/src/ontology).

3 CURRENT WORK

Due to the huge amount of data generated at an increasingly rapid pace, the genomics community is trying to streamline its data processing efforts. Ontologies are an avenue that lead to this, but even they can become too big and unwieldy in their effort to capture all available data. There are instances where the multitude of information captured is not needed.

The Gene Ontology is one the most widely used ontology and one of the most comprehensive. It covers the molecular functions, biological processes and location in cellular components of gene products, containing more than 40,000 terms. In order to make it more accessible and less resource intensive the Gene Ontology Consortium has created slim version of the ontology. These "GO slims" are smaller versions of GO that contain only a subset of the terms, repre-

* To whom correspondence should be addressed:
lschriml@som.umaryland.edu

senting a general knowledge of a specific field, without going too deep into the hierarchy.

Due to the breadth of its user base the DO team decided to create its own slim files, the DO Cancer Slim (Wu *et al.,* 2015) being the most prominent, containing terms needed by the pan-cancer community. In the same vein a DO MGI slim is being created. It contains all the terms that were modified or created during an intensive curatorial effort to map concepts between DO and OMIM. It will give insight into the overlap between DO and OMIM, as well as which disease types are more heavily featured in the MGD. Meaning it can give even more information about which diseases do not have a mouse model to represent them.

4 UPCOMING WORK

Future plans include the definition of all disease terms in DO and the creation of DO slims for every major curation project of all the MODs. These slims will enable DO users to review the representation and classification of MOD associated diseases, to compare the diseases represented between MODs and to compare the different animal models associated with a particular disease or types of diseases across species.

ACKNOWLEDGEMENTS

This work was supported in part by the National Institute of Health – National Center for Research Resources (R01RR025341) and NIH/NIGMS (R01 GM 089820-06).

REFERENCES

Alexandrescu,A. (2001) *Modern C++ Design: Generic Programming and Design Patterens Applied.* Addision Wesley Professional, Boston.

Amberger,J., Bocchini,C. and Hamosh,A. (2011) A new face and new challenges for Online Mendelian Inheritance in Man (OMIM). *Hum. Mutat.*, **32**, 564–567

Bodenreider,O. (2004) The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res.*, **32**, D267–D270.

Donnelly,K. (2006) SNOMED-CT: the advanced terminology and coding system for eHealth. *Stud. Health Technol. Inform.*, **121**, 279–290.

Fang, H. and Gough,J. (2013) DcGO: database of domain-centric ontologies on functions, phenotypes, diseases and more. *Nucleic Acids Res.* **41**:D536-D544.

Köhler,S., Doelken,S.C., Mungall,C.J., Bauer,S., Firth,H.V., Bailleul-Forestier,I., Black,G.C., Brown,D.L., Brudno,M., Campbell,J., FitzPatrick,D.R., Eppig,J.T., Jackson,A.P., Freson,K., Girdea,M., Helbig,I., Hurst,J.A., Jähn,J., Jackson,L.G., Kelly,A.M., Ledbetter,D.H., Mansour,S., Martin,C.L., Moss,C., Mumford,A., Ouwehand,W.H., Park,S.M., Riggs,E.R., Scott,R.H., Sisodiya,S., Van Vooren,S., Wapner,R.J., Wilkie,A.O., Wright,C.F., Vulto-van Silfhout,A.T., de Leeuw,N., de Vries,B.B., Washingthon,N.L., Smith,C.L., Westerfield,M., Schofield,P., Ruef,B.J., Gkoutos,G.V., Haendel,M., Smedle, D., Lewis,S.E. and Robinson,P.N. (2014) The Human Phenotype Ontology project: linking molecular biology and disease through phenotype data. *Nucleic Acids Res.* **4**:D966-D974.

Maiella,S., Rath,A., Angin,C., Mousson,F. and Kremp,O. (2013) Orphanet and its consortium: where to find expert-validated information on rare diseases. *Rev. Neurol. (Paris)*, **169**, S3–S8.

Osborne,J.D., Flatow,J., Holko,M,. Lin,S.M., Kibbe,W.A., Zhu,L.J., Danila,M.I., Feng,G. and Chisholm,R.L. (2009) Annotating the human genome with Disease Ontology. *BMC Genomics*. **10** Suppl 1:S6.

Peng,K., Xu,W., Zheng,J., Huang,K., Wang,H., Tong,J., Lin,Z., Liu,J., Cheng,W., Fu,D., Du,P., Kibbe,W.A., Lin,S.M. and Xia,T. (2013) The Disease and Gene Annotations (DGA): an annotation resource for human disease. *Nucleic Acids Res.* **41**:D553-D560.

Schofield,P.N., Gkoutos,G.V., Gruenberger,M., Sundberg,J.P. and Hancock, J.M. (2010) Phenotype ontologies for mouse and man: bridging the semantic gap. *Dis. Model Mech.,* **3**:281–289.

Schriml,L.M., Arze,C., Nadendla,S., Chang,Y.W., Mazaitis,M., Felix,V., Feng,G. and Kibbe,W.A. (2012) Disease Ontology: a backbone for disease semantic integration. *Nucleic Acids Res.*, **40**, D940–D946.

Singleton MV, Guthery SL, Voelkerding KV, Chen K, Kennedy B, Margraf RL, Durtschi J, Eilbeck K, Reese MG, Jorde LB, Huff CD, Yandell M (2014) Phevor combines multiple biomedical ontologies for accurate identification of disease-causing alleles in single individuals and small nuclear families. *Am. J. Hum. Genet.,* **94**:599-610.

Sioutos,N., de Coronado,S., Ha.ber,M.W., Hartel,F.W., Shaiu,W.L. and Wright,L.W. (2007) NCI Thesaurus: a semantic model integrating cancer-related clinical and molecular information. *J. Biomed. Inform.*, **40**, 30–43.

Vita,R., Overton,J.A., Greenbaum,J.A., Ponomarenko,J., Clark,J.D., Cantrell,J.R., Wheeler,D.K., Gabbard,J.L., Hix,D., Sette,A. and Peters,B (2014) The immune epitope database (IEDB) 3.0. *Nucleic Acids Res.* **43**:D405-D412

Wu,T.J., Shamsaddini,A., Pan,Y., Smith,K., Crichton,D.J., Simonyan,V. and Mazumder,R. (2014) A framework for organizing cancer-related variations from existing databases, publications and NGS data using a High-performance Integrated Virtual Environment (HIVE). *Database (Oxford)*, bau:022.

Wu,T.J., Schriml,L.M., Chen, Q.R., Colbert,M., Crichton,D.J., Finney,R., Hu,Y., Kibbe,W.A., Kincaid,H., Meerzaman,D., Mitraka,E., Pan,Y., Smith,K.M., Srivastava,S., Ward,S., Yan,C. and Mazumder,R. (2015) Generating a focused view of disease ontology cancer terms for pan-cancer data integration and analysis. *Database (Oxford),* bav:032.

Using Semantics and NLP in the SMART Protocols Repository

Olga Giraldo[1,*] Alexander Garcia[1, 2] and Oscar Corcho[1]
[1] Ontology Engineering Group, Universidad Politécnica de Madrid, Spain
[2]Linkingdata I/O LLC, Fort Collins, Colorado, USA

ABSTRACT

In this poster we present the semantic and NLP layers in the development of our repository for experimental protocols. We have studied existing repositories for experimental protocols as well the experimental protocols themselves. We have identified end-user features across existing repositories; we have also structured the semantics for these documents, defined by an ontology and a Minimal Information model for experimental protocols. In addition, we have built an NLP layer that makes extensive use of semantics. Our integrative approach focuses on facilitating search, retrieval and socialization of experimental protocols. We also focus on facilitating the generation of documents that are born semantics.

1 INTRODUCTION

Experimental protocols are fundamental information structures that support the description of the processes by means of which results are generated in experimental research. Well-structured and accurately described protocols (procesable by humans and machines) should facilitate experimental reproducibility. In this poster we present the semantic and NLP infrastructure that we are putting together for machine procesable protocols; we emphasize in the integration of key components of this infrastructure during the implementation of a repository for experimental protocols. Our components include: **i) The SMART Protocols (SP) Ontology**: this ontology results from the analysis of over 200 experimental protocols in various domains –molecular biology, cell and developmental biology and others. Domain experts also participated in the development of the SP ontology (Giraldo, García, & Corcho, 2014). Using the SP ontology allows us to annotate and generate Linked Open Data (LOD) for existing and *de novo* protocols –protocols to be born semantics. **ii) The Sample Instrument Reagent Objective (SIRO) model**. This is a twofold model; on the one hand it defines an extended layer of metadata for this kind of documents. On the other hand, SIRO is a Minimal Information (MI) model conceived in the same realm as PICO (Booth & Brice, 2004), supporting search, retrieval and classification purposes. SIRO is based on an exhaustive study of over 200 protocols in biochemistry, molecular biology, cell and developmental biology, health care as well as interviews with end users. SIRO includes information elements that were identified as central for describing, searching and sharing protocols. Furthermore, as SIRO is rooted in the content of the document, it defines a score of completeness and reproducibility for experimental protocols. iii) **The NLP engine**. The semantics defined by the SP ontology, SIRO, and several domain ontologies is used by our NLP engine, GATE[1]; thus, facilitating search, retrieval and socialization (SeReSo) over experimental protocols. We have generated rules based on the content of protocols; these rules allow us to identify meaningful parts of speech (PoS).

We have reviewed proposed standards for representing experimental protocols, investigations, experiments, scientific documents, rhetorical structures and annotations. In addition, we have analyzed existing repositories for protocols. Interestingly we have found that there are numerous similarities across these repositories –e.g. business model, end-user features, document management; by the same token, the lack of semantics for experimental protocols and the lack of specific features for this particular type of documents may be seen as a common deficiency in these repositories. This document is organized as follows; in section 2 the semantic components are presented; in this section we also inform on the use of semantics by our NLP engine. Some issues and final remarks are presented in section 3.

2 SEMANTICS PLUS NLP

The combination of semantics and NLP makes it possible to deliver a tool that facilitates the generation of experimental protocols that are to be born semantics –fully annotated, linked to the web of data, with fully identified PoS, procesable by machines as well as by humans. In the same vein, a similar process for existing experimental protocols in formats such as PDF is also supported. Furthermore, searching for queries such as: *"What **bacteria** have been used in protocols for **persister cells isolation**?", "What **imaging analysis software** is used for quantitative analysis of locomotor movements, buccal pumping and cardiac activity on **X. tropicalis**?", "How to prepare the stock solutions of the **H2DCF** and **DHE dyes**?",* is also possible.

We are using the SP ontology; SP aims to formalize the description of experimental protocols, which we understand as domain-specific workflows embedded within documents. SP delivers a structured workflow, document and domain knowledge representation written in OWL DL. For the representation of document aspects we are extending the

* To whom correspondence should be addressed: ogiraldo@fi.upm.es

[1] http://gate.ac.uk/

Information Artifact Ontology (IAO).[2] The representation of executable aspects of a protocol is captured with concepts from P-Plan Ontology (P-Plan) (Garijo & Gil, 2012); we are also reusing EXPO (Larisa N. Soldatova & D., 2006), EXACT (L. N. Soldatova, Aubrey, King, & Clare, 2008) and OBI (Courtot et al., 2008). For domain knowledge, we rely on existing biomedical ontologies. Our ontology-based representation for experimental protocols is composed of two modules, namely SP-document[3] and SP-workflow.[4] In this way, we represent the workflow, document and domain knowledge implicit in experimental protocols. By combining both modules we are delivering a born-semantics self- describing document.

We are also working with the SIRO model; our model breaks down the protocol in key elements that are common to "*all*" laboratory protocols: i) Sample/Specimen (S), ii) Instruments (I), iii) Reagents (R) and iv) Objective (O). SIRO is motivated by minimal information models as well as by the **P**atient/Population/Problem **I**ntervention/Prognostic/Factor/Exposure **C**omparison **O**utcome (PICO) model. For the **sample** it is considered the strain, line or genotype, developmental stage, organism part, growth conditions, pre-treatment of the sample and, volume/mass of sample. For the **instruments** it is considered the commercial name, manufacturer and identification number. For the **reagents** it is considered the commercial name, manufacturer and identification number; it is also important to know the storage conditions for the reagents in the protocol. Identifying the **objective** or goal of the protocol, helps readers to make a decision about the suitability of the protocol for their experimental problem. The four elements are also automatically annotated with existing ontologies and exposed as LOD.

The NLP engine, GATE, uses the semantics defined by the SP ontology and SIRO. We have classified our corpus of protocols according to purpose/objective (e.g. extraction of nucleic acids, DNA amplification and visualization of nucleic acids) and then we transformed them to text. For each protocol, metadata available, reagents, instruments samples, actions and instructions were manually identified. We worked with full sentences to characterize PoS, relations, actions (verbs) and full instructions. Gazetteers and rules were thus generated. The results from our NLP workflow are very granular; for instance, we are able to identify DNA purification reagents, digest reaction reagents, cell disruption instruments, etc. Text like "*plant species*" is identified as sample, so are organisms and parts of organisms. The sentences and PoS where the vocabulary is located are also identified and characterized. For instance, PoS such as "***leaf tissue*** *finely* ***ground*** *using a* ***mortar and pestle****, then aliquoted (1 g) for each extraction*" are identified, characterized and annotated; in this example **sample**, **action**, **cell disruption instrument** are identified and characterized. We are using ANNIE (A Nearly-New Information Extraction) as our information extraction system and JAPE for coding rules.

3 FINAL REMARKS

We have presented the integration of three modules in the development of a repository for experimental protocols. Unlike existing repositories, the SP repository focuses on facilitating the production of semantic protocols, intelligent search and retrieval and social activity over experimental protocols. We have extensively studied existing experimental protocols; key functionalities from these will also been included in our repository. We have also presented the SP ontology, the SIRO model for MI and the use of GATE in our architecture. Our workflow addresses scenarios with PDFs and *de novo* protocols – those born semantics based on the SP ontology. For *de novo* documents we are using the ontology as a template; the resulting instantiated RDF is annotated and the conventional document metadata is extracted. For PDFs we are tuning the NLP workflow for extracting SIRO automatically. Extracting the ***O****bjective* has proven to be a challenging task. Actions e.g. *grind the sample*, usually have well defined grammatical structures; but, the ***O****bjective* of the experimental protocol is usually hidden in a complex prose. We are constantly improving the rules; new documents pertaining to other subdomains in biomedical sciences are added to the corpus; then, the rules are tested. Results are manually evaluated and the rules and gazetteers are consequently enriched.

REFERENCES

Booth, A., & Brice, A. (2004). Formulating answerable questions. In A. B. Booth, A (Eds) (Ed.), *Evidence Based Practice for Information Professionals: A Handbook* (pp. 61-70): London: Facet Publishing.

Courtot, Mélanie., Bug, William., Gibson, Frank., Lister, Allyson L., Malone, James., Schober, Daniel., . . . Ruttenberg, Alan. (2008). *The OWL of Biomedical Investigations* Paper presented at the OWLED workshop in the International Semantic Web Conference (ISWC), Karlsruhe, Germany.

Garijo, Daniel., & Gil, Yolanda. (2012). *Augmenting PROV with Plans in P-PLAN: Scientific Processes as Linked Data.* Paper presented at the The 2nd International Workshop on Linked Science 2012, Boston.

Giraldo, Olga., García, Alexander., & Corcho, Oscar. (2014). *SMART Protocols: SeMAntic RepresenTation for Experimental Protocols*. Paper presented at the 4th Workshop on Linked Science 2014 - Making Sense Out of Data (LISC2014), Riva del Garda, Trentino, Italy. http://ceur-ws.org/Vol-1282/lisc2014_submission_2.pdf

Soldatova, L. N., Aubrey, W., King, R. D., & Clare, A. (2008). The EXACT description of biomedical protocols. *Bioinformatics, 24*(13), i295-303. doi: btn156 [pii]10.1093/bioinformatics/btn156

Soldatova, Larisa N., & D., King Roos. (2006). An ontology of scientific experiments. *journal of the royal society interface, 3*(11), 795–803. doi: 10.1098/rsif.2006.0134

[2] https://code.google.com/p/information-artifact-ontology/

[3] http://vocab.linkeddata.es/SMARTProtocols/sp-documentV2.0.htm

[4] http://vocab.linkeddata.es/SMARTProtocols/sp-workflowV2.0.htm

ChEBI for systems biology and metabolic modelling

Janna Hastings[1,*] Neil Swainston[2], Venkatesh Muthukrishnan[1], Namrata Kale[1], Adriano Dekker[1], Gareth Owen[1], Steve Turner[1], Pedro Mendes[2] and Christoph Steinbeck[1]

[1] Cheminformatics and Metabolism, EMBL -- European Bioinformatics Institute (EMBL-EBI), Hinxton, UK
[2] Manchester Institute of Biotechnology, School of Computer Science, University of Manchester, UK

1 INTRODUCTION

ChEBI (http://www.ebi.ac.uk/chebi) is a curated database and ontology of biologically relevant small molecules. It is widely used as a reference for chemicals in the context of biological data such as protein interactions, pathways, and models (Hastings et al., 2013). As of the last release (May 2015), ChEBI contains 44,263 fully curated entries, each of which is classified within one of the sub-ontologies: chemical entities (classified according to structural features), roles (classified according to biological or chemical mode of action or use in application), and subatomic particles.

Systems biology brings together a wide range of information about cells, genes and proteins, as well as the small molecules that act on and within these biological structures. It gives a holistic perspective aiming to track and eventually simulate the entire functioning of biological systems. One aspect of systems biology is metabolic modelling, which aims to develop metabolic reconstructions. At the whole-genome scale, these are all-encompassing interlinked maps of all known metabolic reaction pathways for a given organism (Thiele et al., 2013). Chemical data from ChEBI, such as molecular formula, chemical structure and ontology relationships, can fruitfully be used in the model building and refining process to improve model accuracy and enhance the representation of metabolism (Swainston et al., 2011).

Within this context, efforts are currently underway to improve ChEBI for systems biology and metabolic modelling. The enhancements include the addition of a library, libChEBI, for comprehensive programmatic access to ChEBI data, which will be widely applicable but with a particular focus on metabolic modelling. It will include the facility to determine relationships between molecules, such as stereochemistry, tautometrism and redox pairings, to calculate important physicochemical properties, such as pKa and the Gibbs free energy of formation, and to harness these facilities in support of developing, merging and expanding metabolic models. The library will be open source, available in several programming languages including Java and Python.

ChEBI will be providing a facility for bulk submission of novel compounds which will be automatically classified within the ontology. We will also be undertaking curation of the known metabolomes (i.e. all the known metabolites) across four major species (human, mouse, E. coli and yeast).

Finally, we will be introducing into the ChEBI public website novel visualisations of relevance to the systems biology community, such as chemicals in the context of pathways (powered by Reactome, Croft et al., 2011, and MetaboLights, Haug et al., 2013) and models (powered by BioModels, Li et al., 2010).

ACKNOWLEDGEMENTS

ChEBI is funded by the BBSRC, grant agreement number BB/K019783/1 within the “Bioinformatics and biological resources” fund.

REFERENCES

Croft D, O'Kelly G, Wu G, Haw R, et al. (2011) Reactome: a database of reactions, pathways and biological processes. Nucleic Acids Res. 39:D691-7.

Hastings J., de Matos P., Dekker A., Ennis M., Harsha B., Kale N., Muthukrishnan V., Owen G., Turner S., Williams M. and Steinbeck C. (2013) The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013. Nucleic Acids Res. 41:D456-63.

Haug, K., Salek, R. M., Conesa, P., Hastings, J. et al. (2013) MetaboLights—an open-access general-purpose repository for metabolomics studies and associated metadata. Nucl. Acids Res. 41: D781-D786.

Li, C., Donizelli, M., Rodriguez, N., Dharuri H. et al. (2010) BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. BMC Systems Biology 4:92.

Swainston N, Mendes P, et al. (2011) The SuBliMinaL Toolbox: automating steps in the reconstruction of metabolic networks. Journal of Integrative Bioinformatics, 8(2):186

Thiele, I., Swainston, N., Fleming, R.M.T, et al. (2013) A community-driven global reconstruction of human metabolism. Nature Biotechnology 31, 419–425.

Mapping WordNet to the Basic Formal Ontology using the KYOTO ontology

Selja Seppälä [1]*

[1]Department of Philosophy, University at Buffalo, USA

1 INTRODUCTION

Ontologies are often used in combination with natural language processing (NLP) tools to carry out ontology-related text manipulation tasks, such as automatic annotation of biomedical texts with ontology terms. These tasks involve categorizing relevant terms from texts under the appropriate categories. This requires coupling ontologies with lexical resources. Several projects have realized these kinds of mappings with upper-level ontologies that are extended by domain-specific ontologies (Gangemi *et al.*, 2010; Laparra *et al.*, 2012; Niles and Pease, 2003; Pease and Fellbaum, 2010). However, no such resource is available for the Basic Formal Ontology (BFO), which is widely used in the biomedical domain.[1]

We describe and evaluate a semi-automatic method for mapping the large lexical network WordNet 3.0 (WN) to BFO 2.0 exploiting an existing mapping between WN and the KYOTO ontology, which includes an upper-level ontology similar to BFO. Our hypothesis is that a large portion of WN, primarily nouns and verbs, can be semi-automatically mapped to BFO 2.0 types by means of simple mapping rules exploiting another ontology already linked to WN.

2 ONTOLOGICAL AND LEXICAL RESOURCES

The **Basic Formal Ontology (BFO)** is a domain-neutral upper-level ontology (Smith *et al.*, 2012). It represents the types of things that exist in the world and relations between them. BFO serves as an integration hub for mid-level and domain-specific ontologies, such as the Ontology for Biomedical Investigations (OBI) and the Cell Line Ontology (CLO), which thus become interoperable (Smith and Ceusters, 2010). BFO is subdivided into CONTINUANTS (e.g., OBJECTS and FUNCTIONS) and OCCURRENTS (e.g., PROCESSES and EVENTS). Continuants can be either independent (e.g., physical OBJECTS like persons and hearts) or dependent (e.g., the ROLE of a person as a physician and the FUNCTION of a heart to pump blood). The most recent version, BFO 2.0, represents 35 types to which previous versions (BFO 1.0 and BFO 1.1) have been mapped in Seppälä *et al.*, 2014.

WordNet 3.0 is a large lexical network linking over 117000 sets of synonymous English words (synsets) by means of semantic relations; it is widely used in NLP tasks (Fellbaum, 1998). Noun and verb synsets are linked via the hypernym relation.[2] WN 3.0 distinguishes between types and instances, meaning named entities. It also links a subset of synsets to topic domains (e.g., 'medicine') and semantic labels (e.g., the 'noun.artifact' lexicographer file contains "nouns denoting man-made objects"[3]).

The **KYOTO ontology** (hereafter KYOTO) is part of a project aimed at representing domain-specific terms in a computer-tractable axiomatized formalism to allow machines to reason over texts in natural language (Vossen *et al.*, 2010). It links WordNets of different languages to ontology classes, on the basis of a mapping of the English WN to KYOTO. The approximately 2000 classes of KYOTO are subdivided into three layers: (1) The top-most layer is based on the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE-Lite-Plus, version 3.9.7) and OntoWordNet (Gangemi *et al.*, 2003). **DOLCE** shares a number of relevant characteristics with BFO: domain neutrality; bi-partition into 'endurants' (CONTINUANTS) and 'perdurants' (OCCURRENTS); strict hierarchical *is_a* taxonomy; distinction between independent and dependent entities. (2) The second layer is composed of noun and verb synsets constituting a set of Base Concepts (BCs). (3) The third layer contains domain-specific classes (e.g. from the environmental domain).

3 MAPPING METHOD

Our semi-automatic mapping method involves three main steps:

1. Manually creating mappings:
   - from KYOTO to BFO on the basis of existing mappings of DOLCE to BFO 1.0 and BFO 1.1 (Grenon, 2003; Khan and Keet, 2013; Seyed, 2009; Temal *et al.*, 2010), ignoring the axiomatization incompatibilities;
   - from BFO 1.0 and BFO 1.1 to BFO 2.0 on the basis of work in Seppälä *et al.*, 2014;
   - from WN semantic labels to BFO 2.0.
2. Manually creating mapping rules using the above mappings and extending them with more specific rules from other KYOTO types.
3. Implementing the 33 resulting mapping rules in a Python pipeline using the natural language toolkit for Python that integrates WN 3.0[4] (NLTK 3.0).

The rules are of the form: `'KYOTO/WN > BFO 2.0'`, for example:

```
'#non-agentive-social-object > disposition'
'accomplishment > process'
'noun.act > process'
```

The implementation first lists all KYOTO types that subsume a WN synset using the WN-KYOTO mapping data files.[5] For example, the synset `immunity.n.02` is linked to:

*To whom correspondence should be addressed: seljamar@buffalo.edu

[1] See http://ifomis.uni-saarland.de/bfo/users.

[2] Adjectives and adverbs are linked by way of other semantic relations.

[3] See http://wordnet.princeton.edu/man2.1/lexnames.5WN.html.

[4] Natural Language Toolkit for Python (NLTK), version 3.0, http://www.nltk.org.

[5] http://kyoto-project.eu/xmlgroup.iit.cnr.it/kyoto/index9c60.html?option=com_contentview=articleid=429Itemid=156

```
'Kyoto#condition__status-eng-3.0-13920835-n',
'Kyoto#state-eng-3.0-00024720-n',
'ExtendedDnS.owl#situation',
'ExtendedDnS.owl#non-agentive-social-object',
'ExtendedDnS.owl#social-object',
'DOLCE-Lite.owl#non-physical-object',
'DOLCE-Lite.owl#non-physical-endurant',
'DOLCE-Lite.owl#endurant',
'DOLCE-Lite.owl#spatio-temporal-particular',
'DOLCE-Lite.owl#particular'
```

Second, the mapping rules are applied starting from the more specific ones (BFO leaf nodes): the program tests if a given string (e.g., `'#non-agentive-social-object'`) matches a string in the types list; if the strings match, the program assigns to that synset the corresponding BFO 2.0 type (e.g., `'disposition'`). Thus, the synset `immunity.n.02` is categorized as referring to a subtype of the BFO type DISPOSITION.

4 EVALUATION AND RESULTS

We manually evaluated the method on the 106 synsets in KYOTO marked with a 'medicine' topic domain. 72% of the assigned BFO types were correct (63% of the synsets were assigned the expected BFO type; 8% a superclass). As hypothesized, all the correctly categorized synsets were nominal and verbal. 27% of the assigned BFO types were incorrect (mostly adjectives). One synset was not matched by any rule.

5 DISCUSSION

WN is too large to be manually mapped to BFO. Using the properties of the hypernym hierarchy, we could have approached the problem by mapping the top levels of WN to the relevant BFO types, and propagating the mapped BFO types downwards. However, WN's organization fails to comply with basic ontological principles (Gangemi *et al.*, 2010). Moreover, that method would only cover nouns and verbs, while KYOTO also includes adjectives.

Mapping DOLCE to BFO is not trivial: their categories do not align in every case and are in some cases governed by different axioms. The former is meant to capture our use of language and conceptualization of the world; the latter is a realist ontology and excludes from its scope unicorns and other putative non-real entities. However, these differences will not matter for our purposes here. Mapping WN to BFO is not trivial: WN represents linguistic usage; BFO, entities in the world. WN thus includes synsets that, in BFO terms, do not refer (at all or to a BFO type, e.g. `positive.a.04`). 10 synsets in the evaluation set posed categorization issues.

Our solutions to these issues are: (1) to extend the coverage of the rules by adding other types included in KYOTO and WN's semantic labels; (2) to ignore the axiomatizations. Indeed, this work is neither aimed at mapping DOLCE to BFO, nor at axiomatizing WN. Instead, we attempt to answer the question: to what types of entities do WN synsets refer? The resulting mappings are to be read as 'a WN synset X refers to something that is a subtype of BFO type Y', as in 'the synset `immunity.n.02` refers to a subtype of the BFO type DISPOSITION' — we exclude instances for now. Even a partial mapping should be sufficient to cover a large portion of WN, leaving a smaller subset of problematic cases. An interesting challenge will be to provide BFO-compliant interpretations of unmatched WN synsets.

6 CONCLUSION AND FUTURE WORK

We presented a method to semi-automatically map WordNet 3.0 synsets to BFO 2.0 types via the KYOTO ontology. Our preliminary results are encouraging, but more work is needed to see if the method scales to the full WN. Future work will include: extending the evaluation set of medical synsets using hyponymy relations and other domain resources; carrying out more thorough evaluations, e.g., by randomly extracting samples of synsets grouped by part of speech; augmenting the mapping rules by exploiting other resources, e.g., WN-SUMO mappings and ontologies extending BFO.

ACKNOWLEDGEMENTS

Work on this paper was supported by the Swiss National Science Foundation (SNSF). Thanks also to Christopher Crowner, Barry Smith, and Alan Ruttenberg.

REFERENCES

Fellbaum, C., editor (1998). *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. (2003). Sweetening WordNet with DOLCE. *AI magazine*, **24**(3), 13–24.

Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. (2010). Interfacing WordNet with DOLCE: towards OntoWordNet. In C.-r. Huang, N. Calzolari, and A. Gangemi, editors, *Ontology and the Lexicon: A Natural Language Processing Perspective*, pages 36–52. Cambridge University Press.

Grenon, P. (2003). BFO in a Nutshell: A Bi-categorial Axiomatization of BFO and Comparison with DOLCE. IFOMIS Report 06/2003. Technical report, Institute for Formal Ontology and Medical Information Science (IFOMIS), University of Leipzig, Leipzig, Germany.

Khan, Z. C. and Keet, C. M. (2013). Addressing issues in foundational ontology mediation. In *Proceedings of KEOD'13*, pages 5–16, Vilamoura, Portugal. SCITEPRESS.

Laparra, E., Rigau, G., and Vossen, P. (2012). Mapping WordNet to the Kyoto ontology. In *LREC*, pages 2584–2589.

Niles, I. and Pease, A. (2003). Linking Lexicons and Ontologies: Mapping Wordnet to the Suggested Upper Merged Ontology. In *Proceedings of the IEEE International Conference on Information and Knowledge Engineering*, pages 412–416.

Pease, A. and Fellbaum, C. (2010). Formal ontology as interlingua: The SUMO and WordNet linking project and global WordNet. In C.-r. Huang, N. Calzolari, and A. Gangemi, editors, *Ontology and the Lexicon: A Natural Language Processing Perspective*. Cambridge University Press.

Seppälä, S., Smith, B., and Ceusters, W. (2014). Applying the Realism-Based Ontology-Versioning Method for Tracking Changes in the Basic Formal Ontology. In *8th International Conference on Formal Ontology in Information Systems (FOIS 2014)*, Rio de Janeiro, Brazil.

Seyed, A. P. (2009). BFO/DOLCE Primitive Relation Comparison. In *Nature Precedings*.

Smith, B. and Ceusters, W. (2010). Ontological Realism: A Methodology for Coordinated Evolution of Scientific Ontologies. *Applied Ontology*, **5**, 139–188.

Smith, B., Almeida, M., Bona, J., Brochhausen, M., Ceusters, W., Courtot, M., Dipert, R., Goldfain, A., Grenon, P., Hastings, J., Hogan, W., Jacuzzo, L., Johansson, I., Mungall, C., Natale, D., Neuhaus, F., Rovetto, A. P. R., Ruttenberg, A., Ressler, M., and Schulz, S. (2012). *Basic Formal Ontology 2.0: DRAFT SPECIFICATION AND USER'S GUIDE*.

Temal, L., Rosier, A., Dameron, O., and Burgun, A. (2010). Mapping BFO and DOLCE. *Studies In Health Technology And Informatics*, **160**(Pt 2), 1065–1069.

Vossen, P., Rigau, G., Agirre, E., Soroa, A., Monachini, M., and Bartolini, R. (2010). KYOTO: an open platform for mining facts. In *Proceedings of the 6th Workshop on Ontologies and Lexical Resources*, pages 1–10.

Representing bioinformatics datatypes using the OntoDT ontology

Panče Panov [1]*, Larisa Soldatova [2] and Sašo Džeroski [2]

[1]Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
[2]Brunel University London, Department of Computer Science, Kingston Lane, UB8 3PH, Uxbridge, UK

1 INTRODUCTION

Data processing is at the heart of science. Hence, the problem of data typing is an important problem that has been addressed from different aspects and in different forms. For example, the Research Data Alliance[1] (RDA), whose major goal is to speed up the international data-driven innovation and discovery by facilitating research data sharing and exchange, has identified that the problem of data typing is an important problem that deserves attention. For this purpose, the RDA formed a Data Type Registry (DTR) working group with the goal to: compile a set of use cases for datatype use and management, formulate a data model and expression for datatypes, design a functional specification for type registries, and propose a federation strategy among multiple type registries.

In data mining research it is impossible to efficiently connect parts of workflows (semi-) automatically, such as data pre-processing and data mining, perform analysis of the research results and communicate the research outputs, without machine-processable representation of datatypes and their properties. Hence, there is a need for a standardized semantically-defined and machine amenable representation of scientific datatypes to support cross-domain applications. Unfortunately, the existing representations of datatypes do not fully address such a need. To address this gap, we built an generic ontology for the representation of scientific knowledge about datatypes, named OntoDT (Panov *et al.*, 2015).

2 ONTODT: ONTOLOGY OF DATATYPES

OntoDT defines the meaning of the key entities and represents the knowledge about datatypes in a machine friendly way. The OntoDT ontology is based on the latest revised version of the ISO/IEC 11404[2] standard for datatypes. The design of the OntoDT ontology follows best practices in ontology engineering, such as the OBO Foundry principles. We used the Information Artifact Ontology[3] (IAO) to define the upper level classes and re-used existing ontological resources, such as Open Biomedical Ontologies.

The OntoDT ontology defines the basic entities (see Fig. 1), such as datatype, properties of datatypes, value space, and characterizing operations. We also define a taxonomy of datatypes. The top-level ontology classes include primitive datatypes, generated datatypes, and user defined datatypes. *Primitive datatypes* are defined by explicit specification and are independent of other datatypes. *Generated datatypes* are syntactically and semantically dependent on other datatypes, and are specified implicitly with *datatype generators*. *User defined datatypes* are defined by a datatype declaration and allow defining additional identifiers and refinements to both primitive and generated datatypes. At the lower levels, the datatypes are distinguished with respect to their datatype properties.

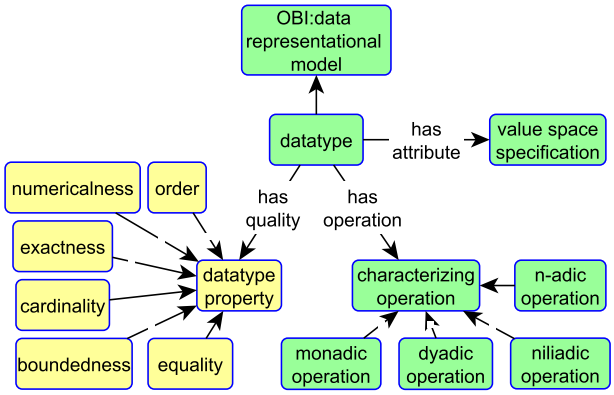


Fig. 1. Representation of datatypes in OntoDT.

OntoDT was used within an Ontology of core data mining entities for constructing taxonomies of datasets, data mining tasks, generalizations and data mining algorithms (Panov *et al.*, 2014). Furthermore, OntoDT can be used for annotation and querying machine learning dataset repositories. OntoDT can also improve the representation of datatypes in the BioXSD exchange format for basic bio-informatics types of data. The generic nature of OntoDT enables it to support a wide range of other applications, especially in combination with other domain specific ontologies: the construction of data mining workflows, annotation of software and algorithms, semantic annotation of scientific articles, etc. OntoDT is open source and is available at `http://www.ontodt.com/` and at BioPortal (`http://bioportal.bioontology.org/`).

3 BIOINFORMATICS DATATYPES

OntoDT is a generic ontology and it allows easy extensions to represent domain specific datatypes. This can be done by directly extending the OntoDT datatype taxonomy and defining the semantic meaning of the domain datatypes by linking them to the corresponding entities in other domain ontologies. For example, we can define an *amino-acid sequence datatype* as a subclass of the *character sequence datatype* class (which is a sequence datatype having characters as its base type). Its semantic meaning can be defined via the IS-ABOUT relation to the *amino acid sequence* entity

*To whom correspondence should be addressed: pance.panov@ijs.si

[1] `http://rd-alliance.org/`

[2] `http://tinyurl.com/qdua9f7`

[3] `http://tinyurl.com/nmjnlw2`

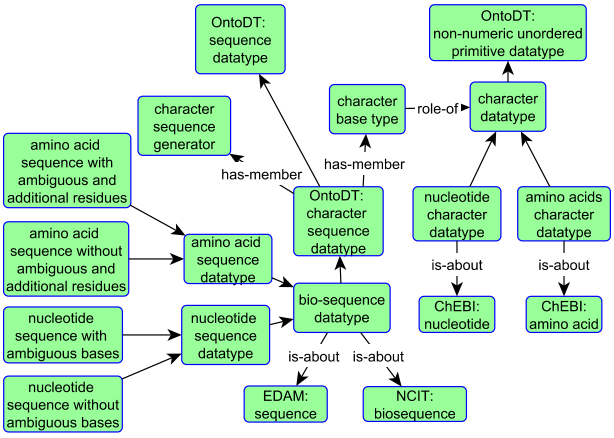


Fig. 2. Representation of bio-sequence datatype from BioXSD in OntoDT.

provided by the National Cancer Institute Thesaurus[4]. In this way, OntoDT can be used for representation of bioinformatics datatypes.

Currently, BioXSD is used to define the basic bio-informatics types of data (Kalaš *et al.*, 2010). BioXSD does not support arbitrary datatypes and it does not provide a clear framework for the representation of the semantic meaning of the data. We propose to enhance the representation of bioinformatics datatypes by exploiting the rigorous taxonomy of datatypes defined in OntoDT and the framework for the representation of semantic meanings adopted by OntoDT. OntoDT is fully interoperable with OBO bio-ontologies because it was developed by following the OBO Foundry recommendations and therefore it fully supports the representation of the semantic meaning of the data by the corresponding entities defined in domain-specific bio-ontologies.

For example, the BioXSD datatype *sequence* represents a string of 1-letter coded nucleotides or amino-acids. A *sequence record* is a datatype containing a sequence, and optionally some metadata about the sequence (for the purpose of identification). The semantic meanings of the terms *sequence* and *nucleotide* are curtail for the capturing of the semantic meaning of the data of the datatype *sequence*. However, the *sequence* datatype is not explicitly linked to the classes *nucleotide* and *amino-acid* defined in the ChEBI ontology[5], recommended by OBO Foundry as a reference ontology.

In Fig. 2, we present the extension of OntoDT to represent the bio-sequence datatype from BioXSD. We represent the *bio-sequence datatype* class as a subclass of the *character sequence datatype* class with the defined semantic meaning in the NCI Thesaurus and the EDAM ontology[6]. In order to define the *nucleotide* and *amino acid sequences datatypes*, we define two subclasses of the *character datatype* class: *nucleotide character datatype* and *amino acid character datatype*. In order to define their semantic meaning, we explicitly link them to the *nucleotide* and *amino acid* classes from the ChEBI ontology. Consequently, the *bio-sequence datatype* class has two subclasses: *nucleotide sequence datatype* and *amino acid sequence datatype*. Furthermore, both datatypes have two subclasses, depending on whether they include ambiguous bases (in the case of nucleotides) or ambiguous and additional residues (in the case of amino acids). For example, the *nucleotide sequence datatype* class has two subclasses: *nucleotide sequence with ambiguous bases* (general nucleotide sequence in BioXSD) and *nucleotide sequence without ambiguous bases* (nucleotide sequence in BioXSD).

In a simmilar way, we represent the *bio-sequence record datatype* class as a subclass of the *record datatype* class. This datatype is defined by a *record generator* and the *bio-sequence-field-list*. As defined in BioXSD, the datatype contains a bio-sequence as a mandatory component and a set of metadata (such as name, note, species, translationalData, reference, inlineBaseQuality) as non-mandatory components. In OntoDT, we model the *bio-sequence field component* class is as a role of the bio-sequence datatype.

BioXSD uses a combined approach of a pure XML Schema annotated by a data-type ontology using Semantic Annotations for Web Services Description Language[7] (WSDL) and XML Schema. SAWSDL defines a set of extension attributes for the WSDL and XML Schema definition languages. Application of attributes allows the description of additional semantics by using references to conceptual semantic models, e.g., ontologies. BioXSD datatypes are annotated with terms from the EDAM ontology Ison *et al.* (2013) using SAWSDL. In the same way, BioXSD datatypes can be annotated with OntoDT terms. For example, by annotating the datatype bio-sequence record from BioXSD with terms from the OntoDT ontology, the web services using this format would have the information that bio-sequence record is in fact a record datatype that is heterogeneous and has components, its values are unordered, it has fixed size, and each component can be accessed by keying.

4 CONCLUSION

The use case presented in this extended abstract demonstrates that OntoDT provides logically consistent representation of bioinformatics datatypes from BioXSD and enables an accurate representation of the semantic meanings of the data of specified datatypes. OntoDT has been designed as a generic and comprehensive ontology of datatypes and consequently any datatype from other resources can also be represented by OntoDT. We suggest that OntoDT can serve as a reference model for the consistent representation of datatypes used within biomedical domains and wider.

REFERENCES

Ison, J., Kalas, M., Jonassen, I., Bolser, D., Uludag, M., McWilliam, H., Malone, J., Lopez, R., Pettifer, S., and Rice, P. (2013). EDAM: an ontology of bioinformatics operations, types of data and identifiers, topics and formats. *Bioinformatics*, **29**(10), 1325–1332.

Kalaš, M., Puntervoll, P., Joseph, A., Bartaševičiute, E., Topfer, A., Venkataraman, P., Pettifer, S., Bryne, J. C., Ison, J., Blanchet, C., Rapacki, K., and Jonassen, I. (2010). BioXSD: the common data-exchange format for everyday bioinformatics web services. *Bioinformatics*, **26**(18), i540–i546.

Panov, P., Soldatova, L., and Džeroski, S. (2014). Ontology of core data mining entities. *Data Mining and Knowledge Discovery*, **28**(5–6), 1222–1265.

Panov, P., Soldatova, L., and Džeroski, S. (2015). Generic ontology of datatypes. *Information Sciences*. (accepted for publication).

[4] http://ncit.nci.nih.gov/

[5] http://www.ebi.ac.uk/chebi/

[6] http://edamontology.org/

[7] http://www.w3.org/TR/sawsdl

Visualization and editing of biomedical ontology alignments in AgreementMakerLight

Catarina Martins [1], Daniel Faria [2], Catia Pesquita [1]

[1]LaSIGE, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa
[2]Instituto Gulbenkian de Ciência

ABSTRACT

Biomedical ontologies pose interesting challenges to the visualization of ontology alignments due to their size and complexity. AgreementMakerLight (AML) is a lightweight ontology alignment system that is particularly suited to the alignment of biomedical ontologies.

Here, we present the updates and evolution of the AML graphical user interface, with a focus on alignment visualization and alignment editing.

1 INTRODUCTION

Several biomedical ontologies have overlapping or related domains, and matching them would greatly increase their interoperability. Ontology matching techniques produce an alignment between two ontologies by establishing correspondences between their elements. Each correspondence is called a mapping and an alignment corresponds to the set of all mappings.

Biomedical ontologies pose challenges in ontology alignment and alignment visualization due to their usually large size, and complexity which can lead to several computational and visualization issues.

AgreementMakerLight(AML) is a lightweight ontology matching system that is particularly well-suited to matching biomedical ontologies, since it can handle large ontologies with complex terminology (Faria *et al.*, 2013a). AML has achieved top performances in the biomedical ontologies tasks in OAEI 2013 (Faria *et al.*, 2013b) and 2014 (Faria *et al.*, 2014), an international competition for ontology alignment systems. It includes several matching techniques supported by a graphical user interfacePesquita *et al.* (2014).

Other ontology matching systems provide user interfaces and visualization and editing features (e.g.: COMA 3.0 (Massmann *et al.*, 2011), AgreementMaker (Cruz *et al.*, 2009), RepOSE (Ivanova and Lambrix, 2012)). However, they struggle to handle large ontologies with multiple inheritance (which is a common case in the biomedical domain).

We present the latest advancements in the graphical user interface for AML, focusing on the novel user alignment editing capabilities and element inspection views. Editing is accompanied by a mapping graph-based visualization that supports users in decision making.

AML is open-source and freely available (as runnable Jar and Eclipse Project) at https://github.com/AgreementMakerLight/AML-Project. For more information, please check: http://aml.fc.ul.pt.

2 AGREEMENTMAKERLIGHT GUI

The graphical user interface of AML comprises three main areas: the Resource Panel where information about the ontologies and the alignment is displayed, like the number of classes, properties and mappings; a Mapping Viewer dedicated to the graph representation of each mapping and its neighbors (Figure 1) and the Alignment Reviewer that lists all the mappings involved in the alignment with information about each one (Figure 2).

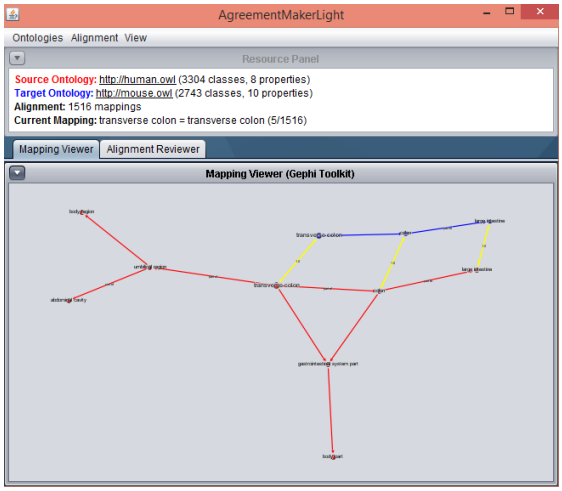


Fig. 1. Visualization of a mapping between two different ontologies in the Mapping Viewer tab.

3 COMPUTING OR LOADING AN ALIGNMENT

The user can load the ontologies in either OWL or RDFs, then he has the option to load a precomputed alignment or to match the ontologies he desires to analyze. In GUI-mode, AML provides two matchers: an automatic matcher and a custom matcher where the user can decide which techniques will be involved in the alignment. The user also has the possibility to repair an alignment (Santos *et al.*, 2013) or to evaluate an alignment against a reference standard . All of these features grant the user the opportunity to save the produced alignment in RDF or in a tab-separated text file.

4 EDITING AN ALIGNMENT

The new update allows the user to alter an existing alignment (either loaded or computed by AML) in the Alignment Reviewer tab. To remove an existing mapping, the user can select it from the list of mappings (Figure 2). To add a new mapping, the user can select the appropriate option and then use a label based search for the classes or properties to map (Figure 3). Both types of changes are recorded when the alignment is saved.

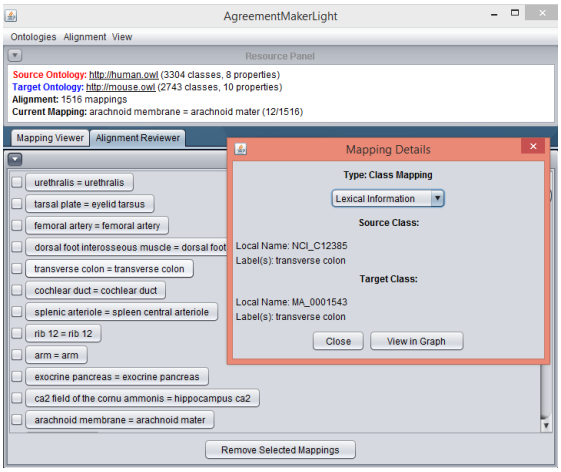


Fig. 2. List of mappings between two different ontologies in the Alignment Reviewer tab.

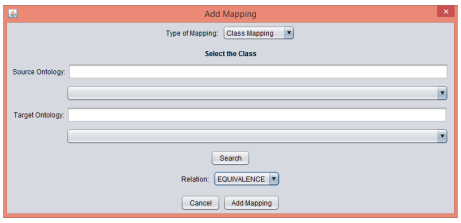


Fig. 3. Add Mapping window in AML.

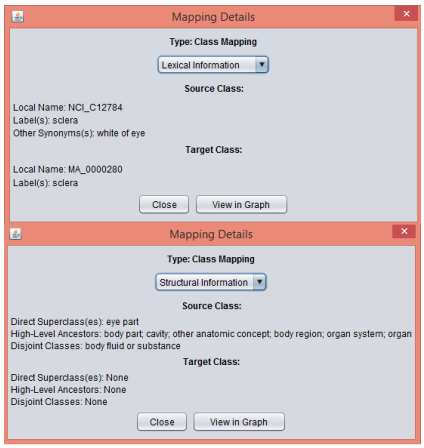


Fig. 4. Class inspection views.

These tasks can be supported by the inspection of each mapping which is accessible from the Alignment Reviewer tab (see Figure 4). The inspection view for classes provides lexical information, which includes local name and synonyms, and structural information, which includes direct superclasses, high-level ancestors and disjoint axioms. The inspection view for properties includes label, domain, range and property type.

5 VISUALIZING A MAPPING

The alignment can be navigated using three different strategies:

- the next/previous mapping option;
- select a mapping from the list of mappings in the Alignment Reviewer tab or in the appropriate sub-menu;
- searching a certain mapping containing a certain term of interest, which is supported by an auto-complete function.

Once a mapping is selected, it can be visualized in the Mapping Viewer tab which includes a graph-based representation of the mapping and its neighborhood. The neighborhood of a mapping includes the classes that are at a predefined distance from the mapped classes, and any mappings between them (see Figure 1).

6 CONCLUSION

User involvement in ontology matching is greatly influenced by the availability of suitable user interfaces and adequate visualization approaches. The recent updates to AgreementMakerLight's user interface have made it possible for users to edit a loaded or computed alignment, while being supported by element inspection capabilities and graph-based visualization of mappings in their context. In future work, we plan to include the visualization of conflicting mappings caused by logical incoherence (Martins *et al.*, 2015). This will allow user to tailor an alignment to their specific purposes since ensuring absolute coherence can decrease the usefulness of an alignment in some cases, due to the loss of meaningful mappings through the repair process (Pesquita *et al.*, 2013).

ACKNOWLEDGEMENTS

This work was partially supported by FCT through funding of LaSIGE Research Unit, ref.UID/CEC/00408/2013.

REFERENCES

Cruz, I. F., Antonelli, F. P., and Stroe, C. (2009). Agreementmaker: efficient matching for large real-world schemas and ontologies. *Proceedings of the VLDB Endowment*, **2**(2), 1586–1589.

Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I. F., and Couto, F. M. (2013a). The agreementmakerlight ontology matching system. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pages 527–541. Springer.

Faria, D., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2013b). Agreementmakerlight results for oaei 2013. page 101.

Faria, D., Martins, C., Nanavaty, A., Taheri, A., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2014). Agreementmakerlight results for oaei 2014.

Ivanova, V. and Lambrix, P. (2012). A system for debugging taxonomies and their alignments. In *Proceedings of the 1st International Workshop on Debugging Ontologies and Ontology Mappings*, volume 79, pages 37–42.

Martins, C., Jimenez-Ruiz, E., Santos, E. P., and Pesquita, C. (2015). Towards visualizing the mapping incoherences in bioportal.

Massmann, S., Raunich, S., Aumüller, D., Arnold, P., and Rahm, E. (2011). Evolution of the coma match system. volume 49.

Pesquita, C., Faria, D., Santos, E., and Couto, F. M. (2013). To repair or not to repair: reconciling correctness and coherence in ontology reference alignments. page 13.

Pesquita, C., Faria, D., Santos, E., Neefs, J.-M., and Couto, F. M. (2014). Towards visualizing the alignment of large biomedical ontologies. In *Data Integration in the Life Sciences*, pages 104–111. Springer.

Santos, E., Faria, D., Pesquita, C., and Couto, F. (2013). Ontology alignment repair through modularization and confidence-based heuristics. *arXiv preprint arXiv:1307.5322*.

Inferring logical definitions using compound ontology matching

Daniela Oliveira* and Catia Pesquita

LaSIGE, Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa, Campo Grande 1749-016, Portugal

ABSTRACT

OBO logical definitions are a means to support the creation of integrated reference ontologies. In ontologies they exist for, logical definitions currently cover a small portion of classes, which limits the potential for integration.

We present a novel preliminary strategy to derive logical definition candidates based on an ontology compound matching algorithm. Preliminary results show that this strategy is able to increase the coverage of logical definitions between 2 and 19%.

1 INTRODUCTION

The Open Biological and Biomedical Ontologies (OBO) Foundry (Smith *et al.*, 2007) is a collaborative initiative for establishing a set of principles for ontology development in the biomedical domain. Its goal is to support the creation of orthogonal interoperable reference ontologies and OBO cross-products were created to provide computable logical definitions for classes.

Several of the current logical definitions present in the OBO Foundry were obtained with the Open Bio-Ontology Language (Obol) (Mungall, 2004). Obol has a fairly complex set of rules to define ontology-specific grammars and generate potential logical definitions, which have to be manually curated. It has been applied in the improvement of phenotype ontologies (Mungall *et al.*, 2010) and in the normalization of GO (Mungall *et al.*, 2011). A more recent approach, cross-products extension (CPE) (Quesada-Martínez *et al.*, 2014) has been applied to the GO.

However, adding and maintaining these definitions requires a significant amount of effort, which likely contributes to their incomplete coverage. For instance, the logical definitions of the three ontologies employed in this paper account for less than half of the classes in the ontology (see Table 1).

Ontology	Classes	Logical Definitions	Proportion
HP	28621	14059	49.1%
MP	28643	7694	26.9%
WBP	2290	957	41.7%

Table 1. Proportion of classes represented by logical definitions.

This paper describes a preliminary strategy to derive logical definitions candidates that is based on a novel algorithm used for the creation of compound alignments. Our algorithm is centered around a ternary compound mapping approach, which we define as a tuple <X, Y, Z, R, M>, where X, Y and Z are classes from three distinct ontologies, R is a relation established between Y and Z to generate a class expression that is mapped to X via a mapping relation M. Here, we consider the ontology to which X belongs to be the source ontology, and the ontologies that define Y and Z to be the target ontology 1 and 2, respectively. In this particular case the relation R is always an intersection and the mapping M an equivalence.

```
[Term]
id: HP:0000337 ! broad forehead
intersection_of: PATO:0000600 ! increased width
intersection_of: inheres_in FMA:63864 ! forehead
```

Fig. 1. Example of a possible ternary compound match in the HP logical definitions.

Due to the nature of the matching algorithm our strategy only finds logical definitions for classes which are composed of constructs from two different ontologies. This is the case of many of the classes in the Human Phenotype Ontology which have definitions that are composed of classes from the PATO and FMA ontologies (see Figure 1). Our goal is to investigate whether our proposed strategy is able to reliably find definitions which were not obtained through previous methodologies, and where thus not included in the available logical definitions.

2 MATERIALS AND METHODS

2.1 Ontologies

For creating and testing our algorithm we matched different combinations (see Table 2) of the following OBO ontologies: Cell Type (CL) (Bard *et al.*, 2005), Foundational Model of Anatomy (FMA) (Rosse and Mejino, 2003), Gene Ontology - Biological Process (GO) (Ashburner *et al.*, 2000), Human Phenotype Ontology (HP) (Köhler *et al.*, 2013), Mammalian Phenotype (MP) (Smith *et al.*, 2004), Neuro Behaviour Ontology (NBO) (Gkoutos *et al.*, 2012), Phenotypic quality (PATO) (Mungall *et al.*, 2010), Uber Anatomy Ontology (UBERON) (Haendel *et al.*, 2009) and *Caenorhabditis elegans* phenotype (WBP) (Schindelman *et al.*, 2011).

These ontologies were downloaded from the OBO Foundry (http://obo.sourceforge.net) in February 2015.

2.2 Algorithm

We developed a novel algorithm (Oliveira and Pesquita, 2015) to establish compound mappings integrated in AgreementMakerLight (AML) (Faria *et al.*, 2014) ontology matching system. We compute the confidence of the first step, based on the ratio of words of the first target ontology classes' labels that overlap with the words of the labels of the classes of the source ontology, weighted by their evidence content (i.e., the inverse log of their frequency in the

*To whom correspondence should be addressed: doliveira@lasige.di.fc.ul.pt

ontology's vocabulary). In the second step, we filter out source classes whose matches were below the threshold, and then match the remaining ones based on their unmatched words in step 1, to the second target ontology. To compute the confidence of this second step, if the number of words of a certain label is higher than the number of words of a target 2 ontology label we compare the unmatched words to the each word of the target 2 terms. Else, if the number of words of a certain label is lower than the number of words of a target 2 ontology label we compare the unmatched words to the each word of the source. Finally, the algorithm had a greedy selection step, which selects the mapping with the highest similarity, amongst the source classes with more than one mapping.

2.3 Evaluation

To evaluate our strategy we performed a manual analysis of the results, where we classified mappings into three possible categories: 'Correct', where the mapping is deemed correct and the source class has no mapping in the logical definitions; 'Conflict', where the mapping is potentially correct but the source class has a different mapping in the logical definitions; and 'Incorrect', where the mapping is deemed incorrect. We applied this to all mappings created by using 0.5 as a threshold for step 1 and 0.9 for step 2.

3 RESULTS AND DISCUSSION

The manual evaluations of the mappings (Table 2) reveals a very low proportion of incorrect mappings, and an intermediate proportion of conflicting mappings. Given the low error rate, we consider our strategy to be suitable to the identification of candidate logical definitions. However, we are also interested in ascertaining whether our strategy can contribute with a signficant number of novel definitions. In fact, the novel logical definitions represent a percentual increase between 2 and 19%, which corresponds to more than 800 new logical definitions for the three ontologies (see Table 3). This indicates that our strategy is able to find candidate logical definitions which are missed by the currently employed methods.

	Correct	Conflict	Incorrect
MP-CL-PATO	63.71 %	34.60 %	1.69 %
MP-GO-PATO	92.16 %	6.97 %	0.87 %
MP-NBO-PATO	72.46 %	26.09 %	1.45 %
MP-UBERON-PATO	91.33 %	7.96 %	0.70 %
WBP-GO-PATO	88.55 %	7.49 %	3.96 %
HP-FMA-PATO	77.82 %	15.56 %	6.61 %

Table 2. Manual evaluation of results.

Ontology	New Mappings	Logical Definitions	% of Growth
HP	259	14059	1.84
MP	422	7694	5.48
WBP	182	957	19.02

Table 3. Impact of the new mapping derived logical definitions.

However, for some ontologies, the number of conflicting mappings represents a greater proportion. Upon comparing the novel mapping with the conflicting logical definition we have found that in many cases this is due to similar PATO classes, whose synonyms are hard to distinguish.

4 CONCLUSION

Our proposed strategy was able to successfully identify a significant number of novel logical definitions candidates, with a low error rate. Therefore, this new methodology could help expert curators expand the current logical definitions. Although our current approach is limited to logical definitions established by the intersection of classes from two distinct external ontologies, we expect it can easily be adapted to logical definitions that employ classes from the source ontology and a single external ontology. In the future, we will also explore how different similarity thresholds can affect the accuracy and coverage of the obtained logical definitions.

ACKNOWLEDGEMENTS

The authors are grateful to Daniel Faria for his technical support. This work was supported by FCT through funding of LaSIGE Research Unit, ref.UID/CEC/00408/2013

REFERENCES

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., *et al.* (2000). Gene Ontology: tool for the unification of biology. *Nature genetics*, **25**(1), 25–29.

Bard, J., Rhee, S. Y., and Ashburner, M. (2005). An ontology for cell types. *Genome biology*, **6**(2), R21.

Faria, D., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2014). AgreementMakerLight: A scalable automated ontology matching system. *10th International Conference on Data Integration in the Life Sciences 2014 (DILS)*, page 29.

Gkoutos, G. V., Schofield, P. N., and Hoehndorf, R. (2012). The neurobehavior ontology: an ontology for annotation and integration of behavior and behavioral phenotypes. *Int Rev Neurobiol*, **103**, 69–87.

Haendel, M. A., Gkoutos, G. G., Lewis, S. E., and Mungall, C. (2009). Uberon: towards a comprehensive multi-species anatomy ontology.

Köhler, S., Doelken, S. C., Mungall, C. J., Bauer, S., Firth, H. V., Bailleul-Forestier, I., Black, G. C., Brown, D. L., Brudno, M., Campbell, J., *et al.* (2013). The Human Phenotype Ontology project: linking molecular biology and disease through phenotype data. *Nucleic acids research*, page gkt1026.

Mungall, C. J. (2004). Obol: integrating language and meaning in bio-ontologies. *Comparative and functional genomics*, **5**(6-7), 509–520.

Mungall, C. J., Gkoutos, G. V., Smith, C. L., Haendel, M. A., Lewis, S. E., and Ashburner, M. (2010). Integrating phenotype ontologies across multiple species. *Genome biology*, **11**(1), R2.

Mungall, C. J., Bada, M., Berardini, T. Z., Deegan, J., Ireland, A., Harris, M. A., Hill, D. P., and Lomax, J. (2011). Cross-product extensions of the Gene Ontology. *Journal of biomedical informatics*, **44**(1), 80–86.

Oliveira, D. and Pesquita, C. (2015). Compound matching of biomedical ontologies. In *International Conference on Biomedical Ontology (ICBO)* (to appear).

Quesada-Martínez, M., Mikroyannidi, E., Fernández-Breis, J. T., and Stevens, R. (2014). Approaching the axiomatic enrichment of the Gene Ontology from a lexical perspective. *Artificial intelligence in medicine*.

Rosse, C. and Mejino, J. L. (2003). A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *Journal of biomedical informatics*, **36**(6), 478–500.

Schindelman, G., Fernandes, J. S., Bastiani, C. A., Yook, K., and Sternberg, P. W. (2011). Worm Phenotype Ontology: integrating phenotype data within and beyond the c. elegans community. *BMC bioinformatics*, **12**(1), 32.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., *et al.* (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, **25**(11), 1251–1255.

Smith, C. L., Goldsmith, C.-A. W., and Eppig, J. T. (2004). The Mammalian Phenotype Ontology as a tool for annotating, analyzing and comparing phenotypic information. *Genome biology*, **6**(1), R7.

Modeling and Tools for Supporting Post-Coordination in ICD-11

Csongor I. Nyulas*, Samson W. Tu, Tania Tudorache and Mark A. Musen
Stanford Center for Biomedical Informatics Research, Stanford, CA, USA

ABSTRACT

The World Health Organization (WHO) is working on the 11th revision of the International Classification of Diseases (ICD-11), which is scheduled to be released in 2017. One of the main goals for the revision process was to make ICD-11 ready to be used with electronic health records and information systems. In order to meet this goal, ICD-11 was designed to have an ontological underpinning, commonly referred to as its "Content Model". Another significant novelty introduced in ICD-11, compared to ICD-10, is that ICD-11 will allow medical coders to enter details, such as severity and temporal course, as modifiers to the main ICD code. To support this kind of post-coordination, we extended the ICD-11 Content Model and the collaborative ICD authoring tool, iCAT, so that ICD developers can specify which post-coordination axes are allowed for each ICD-11 entity, to specify valid values for each axis from a special ICD-11 chapter where value sets are defined, called Chapter X, and to give selected ICD-11 categories logical definitions, so that equivalence of multiple codes and already defined categories can be established automatically.

1 INTRODUCTION

The International Classification of Diseases and Related Health Problems (ICD) is "the international standard diagnostic tool for epidemiology, health management and clinical purposes".[1] The current 10th edition of ICD (ICD-10) was endorsed by the World Health Assembly in 1990 and has been updated periodically over the years. ICD-11 is currently developed as a collaborative effort supported by Web-based software, called iCAT (Tudorache *et al.*, 2011). iCAT is a specially configured and enhanced instance of the popular ontology editor WebProtégé (Tudorache *et al.*, 2013). A key to this revision effort is the, so called, *Content Model* (WHO, 2011), designed to support detailed description of the clinical characteristics of each category, clear relationships to other terminologies and classifications, especially SNOMED-CT, multi-lingual development, and sufficient content so that the adaptations for alternative uses cases for the ICD can be generated automatically.

Starting with the availability of the 11th revision of ICD, users of the classification will be able to code diseases and other health-related conditions by using combinations of codes in accordance with a specific grammar, a practice commonly referred as post-coordination. Post-coordination is an effective way to avoid the combinatorial explosion that would result if ICD were to specify codes for all possible manifestations of health conditions (i.e., pre-coordination). To support post-coordination, we have extended the ICD Content Model and the ICD collaborative authoring tool (iCAT) with new features that allow subject-matter experts to edit post-coordination information. In particular, we have added two new tabs to iCAT, the *Post-Coordination* tab and the *Logical Definition* tab.

*To whom correspondence should be addressed: nyulas@stanford.edu

[1] http://www.who.int/classifications/icd/en/

2 MODELING POST-COORDINATION

The first step towards supporting post-coordination in ICD-11 was to extend its ontological foundation, the Content Model, with properties and classes and restrictions between those in a way that allows ICD editors to specify what are all possible ways to post-coordinate each ICD entity.

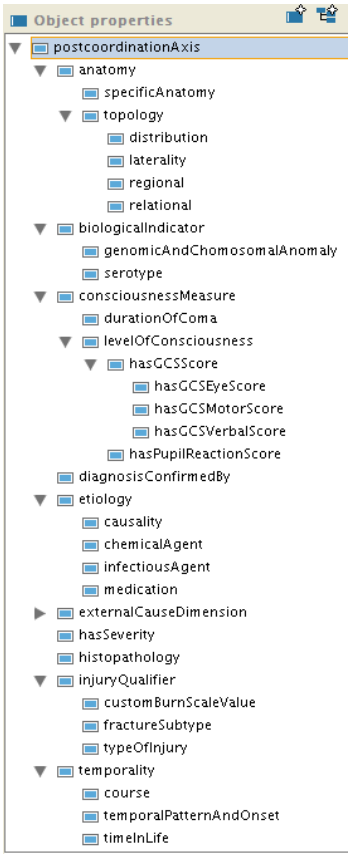


Fig. 1. Screenshot from the Protégé ontology editor, showing the supported post-coordination axes that have been added to the ICD-11 Content Model.

ICD-11 introduces a number of **post-coordination axes** (such as *severity*, *onset*, and *anatomic location*), which we modeled as an object property hierarchy (Figure 1). ICD-11 also introduces a dedicated chapter, **Chapter X**, which will contain possible values for each of these post-coordination axes (Figure 2).

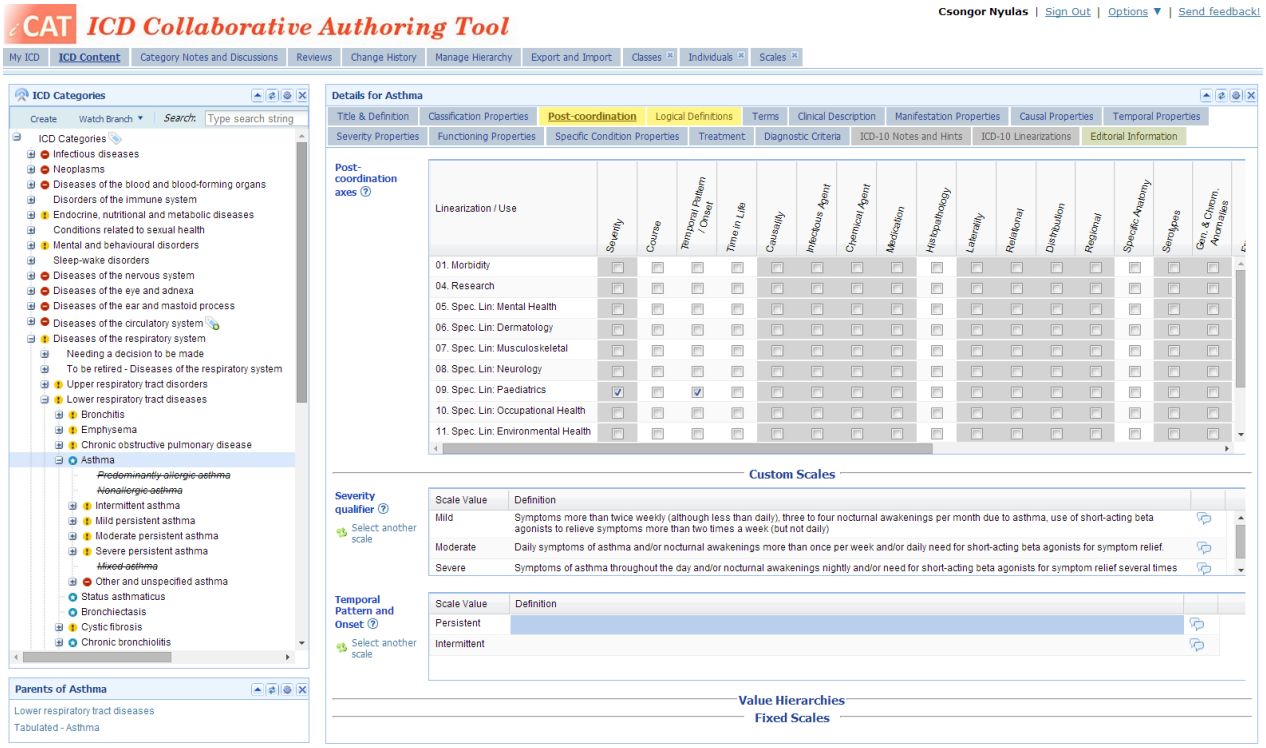


Fig. 3. iCAT screenshot showing how one can specify that "Asthma" can be post-coordinated on "Severity" and "Temporal pattern" axes in a hypothetical "Pediatrics" linearization. Also shows that for the "Severity" axis different values of the "MildModerateSevere" scale may be used. Both severity values ("Mild", "Moderate", "Severe") and temporal-pattern values ("Intermittent" and "Persistent") may have "local" Asthma-specific definition.

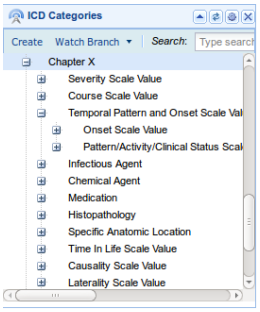


Fig. 2. Partial screenshot of Chapter X in iCAT.

3 EDITING POST-COORDINATION IN iCAT

In addition to modeling changes, we had to extend iCAT in several ways in order to support editing of post-coordination information.

We introduced two new tabs for editing post-coordination details. The ***Post-coordination* tab** allows iCAT users to specify the valid post-coordination axes that may be used with a given entity in each use case, or "linearization", as it is called in ICD terms. In the same tab, the user is also able to specify the possible value scales or value hierarchies that may be used to post-coordinate the entity along a given axis (see Figure 3). The ***Logical Definitions* tab** allows ICD-11 authors to give an existing ICD entity a logically necessary and sufficient definition by selecting a superclass and specifying values for some allowed post-coordination axes of that superclass.

Another modification was to configure iCAT to show appropriate tabs with appropriate content for editing the details for entities in **Chapter X**. Unlike in other chapters, for entities in Chapter X, such as *Chronic*, iCAT users can only edit *title*, *definition*, *synonyms* and *external definitions* in the *Title & Definition* tab, can define its *base index terms* in the *Terms* tab, and specify which linearizations will the entity be part of in the *Classification Properties* tab.

We have also added a top level tab, called ***Scales***, designed for the editing of predefined value scales that can be used to define possible values for the *severity*, *course* and *onset* post-coordination axes. The definition of the scale values in these scales can be overwritten at each ICD entity. This tab is available only to WHO.

ACKNOWLEDGEMENTS

This work is supported in part by NIH grants GM086587 and GM103316. We are also grateful for the generous support of Ms. Marilyn Allen and the Council of Colleges of Acupuncture and Oriental Medicine (CCAOM).

REFERENCES

Tudorache, T., Nyulas, C. I., Noy, N. F., Redmond, T., and Musen, M. (2011). iCAT: A collaborative authoring tool for ICD-11. In *Workshop Ontologies Come of Age in the Semantic Web*, number 809 in CEUR Workshop Proceedings, pages 72–74.

Tudorache, T., Nyulas, C., Noy, N. F., and Musen, M. A. (2013). WebProtégé: A collaborative ontology editor and knowledge acquisition tool for the web. *Semantic Web Journal*, **4**(1).

WHO (2011). The ICD-11 Content Model. `http://www.who.int/classifications/icd/revision/contentmodel/`.

FAIRDOM approach for semantic interoperability of systems biology data and models

Olga Krebs[1], Katy Wolstencroft[3], Natalie Stanford[2], Norman Morrison [2], Martin Golebiewski[1], Stuart Owen[2], Quyen Nguyen[1], Jacky Snoep[2], Wolfgang Mueller[1] , and Carole Goble[2].

[1] Heidelberg Institute for Theoretical Studies, Heidelberg, Germany
[2] School of Computer Science, University of Manchester, UK
[3] Leiden Institute of Advanced Computer Science, Leiden, NL

ABSTRACT

The ability to collect and interlink heterogeneous data and model collections is essential in Systems Biology. Effective data exchange and comparison requires sufficient data annotation. This is particularly apparent in Systems Biology, where data heterogeneity means that multiple community metadata standards are required for the annotation of a whole investigation, including data, models and protocols.

Here we describe FAIRDOM (http://fair-dom.org/) strategy in the context of semantic data management in the openSEEK , a web-based resource for sharing and exchanging Systems Biology data and models.

1 INTRODUCTION

Data integration is an essential part of Systems Biology. Scientists need to combine different sources of information in order to model biological systems, and relate those models to available experimental data for validation. . Currently, only a small fraction of the data and models produced during Systems Biology investigations are deposited for reuse by the community, and only a smaller fraction of that data is standards compliant, semantic content.

By embedding semantic technologies into familiar data management tools, the SEEK [1] enables semantic annotation of new data and the generation and querying of linked-data compliant datasets, whilst hiding the complexities of ontologies and metadata from its users. The SEEK is based on the ISA infrastructure (Investigations, Studies and Assays), a standard format for describing how individual experiments (assays) are aggregated into wider studies and investigations [2]. This poster will present the semantic data integration in SEEK, and how it supports the whole life cycle of data collection, annotation, sharing and reuse of Systems Biology data and resources.

2 THE JERM ONTOLOGY AND JERM TEMPLATES

The JERM Ontology is an application ontology designed to describe the relationships between items in SEEK (for example, data, models, experiment descriptions, samples, protocols, standard operating procedures and publications); and to enable these relationships to be expressed with formal semantics. It is based on the idea of the Minimal Information Models (https://www.biosharing.org), which have been collected under the umbrella of MIBBI (Minimum Information for Biological and Biomedical Investigations). The JERM takes the specification one step further, expressing the minimum information model as an OWL ontology (http://bioportal.bioontology.org/ontologies/JERM).

The majority of laboratory scientists use spreadsheets for the daily management and manipulation of data, so the RightField semantic spreadsheet application [3] (also developed during this work) is used to embed semantic annotation into the data. RightField-enabled spreadsheets allow the collection of semantic information by stealth.

By embedding the JERM metadata model in a spreadsheet format, and enabling the use of JERM (and other) vocabulary terms for annotation, the process of standardized semantic data collection can become part of the existing data management activities in the laboratory. JERM spreadsheet templates have been developed for a wide range of experimental data types.

REFERENCES

1.Wolstencroft, K., Owen, S., du Preez, F., Krebs, O., Mueller, W., Goble, C. and Snoep, J.L. (2011) The SEEK: a platform for sharing data and models in Systems Biology. Methods Enzymol, 500, 629-655.

2. Rocca-Serra, P., Brandizi, M., Maguire, E., Sklyar, N., Taylor, C., Begley, K., Field, D., Harris, S., Hide, W., Hofmann, O. et al.(2010) *ISA software suite: supporting standards-compliant experimental annotation and enabling curation at the community level*. Bioinformatics, 26, 2354-2356.

3. Wolstencroft, K., Owen, S., Horridge, M., Krebs, O., Mueller, W., Snoep, J.L., du Preez, F. and Goble, C. (2011) *RightField: embedding ontology annotation in spreadsheets.* Bioinformatics, 27, 2021-2022

* To whom correspondence should be addressed: olga.krebs@h-its.org

Mapping a Database Schema to the Structure of an Existing Ontology

Anahita Nafissi*, Fabio Fiorani and Björn Usadel
Plant Sciences (IBG-2), Forschungszentrum Jülich, D-52425 Jülich, Germany
* a.nafissi@fz-juelich.de

In this work, we describe an approach for mapping the structure of a database schema to the structure of existing ontologies. The database contains plant traits values and plant experimental history. The goal is to identify the semantic correspondences between databases and ontologies and provide a tool that can be more broadly adopted by the community. The approach presented in this work is a semi-automatic approach.

In the literature, there are several approaches which map the database schema to an ontology. The underlying assumption by all approaches is that the chosen ontologies model the same domain as the one modelled by the relational database schema. Some mapping approaches are R2O (Barrasa et al., 2004), DartGrid (Chen et al., 2006), Linked Data Mapper (Zhou et al., 2008), RDOTE (Vavliakis et al., 2010), RDB2OWL [Bumans & Cerans, 2010], MAPONTO [An et al., 2006]. The difference between the above approaches is that some approaches are manual and some are semi-manual. Furthermore, for some approaches a human expert gives the correspondences between database terms and ontology terms.

The database schema of our Phenomis database considered for mapping contains plant phenotyping information and environmental information. The ontologies considered for mapping are plant ontology, phenotypic quality ontology, plant trait ontology, plant environmental conditions, and environment ontology. The mentioned ontologies are very large and contain over 1000 concepts. Unlike the number of concepts, the number of roles is very small (less than 10). Note that the roles denote the relations between domain objects.
In order to map the database to the ontology, we first consider the schema of the database and extract relation names and attributes of each relation. Note that the relations in a relational schema are classified into two categories, namely entity relations and relationship relations (Hu & Qu, 2007). Furthermore, an attribute is also classified into two categories, namely foreign key attribute and non foreign key attribute (Hu & Qu, 2007). A relationship relation is used to connect two other relations and contains foreign key attributes. Unlike a relationship relation which contains only foreign key attributes, an entity relation contains non foreign key attributes. For the mapping process, we do not consider relationship relations and all their attributes. Similarly, we extract concept - and role names of the ontology.

For the mapping process, we have to discover the correspondences between the terms of the database and the terms of the ontology. For this purpose, we compare the relation - and attribute names of the database with the concept names of the ontology. The comparison is performed according to the similarity matches. This means that we find similar matches among the relation and attribute names of the database and concepts of the ontology. Then, the results should be evaluated by a human expert (plant biologist) who is familiar with both the terms used in the database and in the ontologies. Thus, this approach is a semi-automatic approach. For some ontologies the mapping results are more than the others. Furthermore, the human involvement required for mapping varies across different ontologies.
The softwares used for this work are Java, Protégé, SQL.

ACKNOWLEDGEMENTS

This work is performed within the German-Plant-Phenotyping Network which is funded by the German Federal Ministry of Education and Research (project identification number: 031A053)

REFERENCES

An, Y., Borgida A. and Mylopoulos, J. (2006). Discovering the Semantics of Relational Tables through Mappings, Journal on Data Semantics, VII, pp. 1-32.

Barrasa, J., Corcho O., and Gomez-Perez A (2004). R2O, an Extensible and Semantically Based Database-to-Ontology Mapping Language, in Second International Workshop on Semantic Web and Databases (SWDB 2004).

Bumans, G., and Cerans, K. (2010). RDB2OWL: a Practical Approach for Transforming RDB Data into RDF/OWL, in A. Paschke, N. Henze and T. Pellegrini, eds. Proceedings of the 6th International Conference on Semantic Systems (I-SEMANTICS 2010), ACM.

Chen, H., Wu, Z., Mao, Y., and Zheng, G. (2006). DartGrid: a Semantic Infrastructure for Building Database Grid Applications, Concurrency and Computation: Practice and Experience, 18(14), pp. 1811-1828.

Hu, W., and Qu, Y. (2007). Discovering Simple Mappings between Relational Database Schemas and Ontologies, Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference (ISWC/ASWC2007), Busan, South Korea, Volume 4825 of LNCS, page 225—238, Berlin, Heidelberg, Springe Verlag.

Vavliakis, K.N., Grollios, T.K., and Mitkas, P.A. (2010). RDOTE-Transforming Relational Databases into Semantic Web Data, in A. Polleres and H. Chen, eds., Proceedings of the ISWC 2010 Posters & Demonstrations Track: Collected Abstracts, pp.121-124.

Zhou, C., Xu, C., Chen, H. and Idehen, K. (2008). Browser-based Semantic Mapping Tool for Linked Data in Semantic Web, in C. Bizer, T. Heath. K. Idehen and T. Berners-Lee, eds., Proceedings of the WWW 2008 Workshop on Linked Data on the Web (LDOW 2008).

Part IV

Demo Abstracts

GOfox: Semantics-based simplified hierarchical classification and interactive visualization to support GO enrichment analysis

Edison Ong, Yongqun He
University of Michigan, Ann Arbor, Michigan, USA

ABSTRACT

Gene Ontology (GO)-based statistical enrichment analysis is a popular approach to identify statistically enriched biological processes, molecular functions, and cellular components that are associated with a list of genes. However, such GO enrichment analysis often generates a large number of enriched GO terms that are difficult to interpret and analyze. To address this issue, we developed GOfox, a web tool that utilizes OWL-based ontology semantics and RDF triple store SPARQL queries to generate full or simplified hierarchical GO subsets to classify and display enriched GO terms. GOfox integrates and extends features from OntoFox and Ontobee, two ontology tools developed in the laboratory. GOFox also includes a newly developed algorithm for generating simplified hierarchical classification by considering the multiple inheritance of GO. Furthermore, GOfox provides an interactive visualization that supports GO subset tree exploration and term editing. GOfox is freely available at the website: http://gofox.hegroup.org/.

1 INTRODUCTION

A biological/biomedical ontology is a set of computer and human-interpretable terms and relations that represents entities in a biological/biomedical domain and how they relate to each other. Hundreds of biological ontologies have been developed. The most widely used biological ontology is the Gene Ontology (GO), which systematically and semantically represents three major attributed associated with gene products: Biological Processes (BP), Molecular Function (MF), and Cellular Components (CC) (Ashburner et al., 2000). One major GO application is GO-based statistical enrichment analyses. The rationale of such an enrichment analysis is that given a group of genes, the co-functioning genes should have a higher or enriched potential to be identified as a relevant group using high throughput technologies (*e.g.*, microarrays and RNA-Seq). Since often hundreds (or even more) of enriched terms are detected, the linear output of enriched terms can be very large and overwhelming, resulting in diluted focus on the analysis of related terms.

To address the ever increasing number of enriched GO terms resulting from high throughput studies, we developed GOfox to support GO enrichment analysis through integrating and extending the features of OntoFox (Xiang et al., 2010) and Ontobee (Xiang et al., 2011). OntoFox is able to fetch ontology terms and axioms. OntoFox includes several semantics algorithms for extracting different levels of intermediate layer terms between user-selected terms and a top level term of the ontology (Xiang et al., 2010). Ontobee is the default OBO ontology linked data server that facilitates ontology data sharing, visualization, query, integration, and analysis (Xiang et al., 2011). Ontobee also supports ontology visualization including the hierarchy, definition and annotations. By integrating and extending the features of OntoFox and Ontobee, GOfox is able to represent the enriched GO terms in an interactive hierarchical layout along with term-related information, and it allows users to manually modify the summarized enrichment result. Considering the multiple inheritance strategy used in GO development, GOfox developed a new algorithm to trim down the size of the enriched subset tree of GO. In addition, GOfox retrieves and displays related information such as definition, database cross references and comments, etc. of the selected GO term from Ontobee. This report provides the first time introduction of the GOfox to help researchers better visualize and analyze the results of GO gene enrichment studies.

2 GOFOX SYSTEM OVERALL DESIGN

The overall design and workflow is displayed in Fig. 1. Using a web form shown in Fig. 2, a user can input enriched or interested GO terms along with the p-values. Then the user can define a P-values cutoff (or another cutoff) and how intermediates are treated. After receiving the user's request, the GOfox server will extract a subset of GO that contains the input terms and related GO terms using PHP, Java and SPARQL. Specifically, the server queries against He Group's RDF triple store using SPARQL and retrieves a subset of GO. The query results will be in RDF/XML format and will be reformatted to the OWL format using OWL API (http://owlapi.sourceforge.net/). Then, based on the user's preference, GOfox will run simplification algorithm and generate results for downloading, visualization, and editing (Fig. 1). The results will be temporarily stored in He group RDF triple store and destroyed in a regular basis.

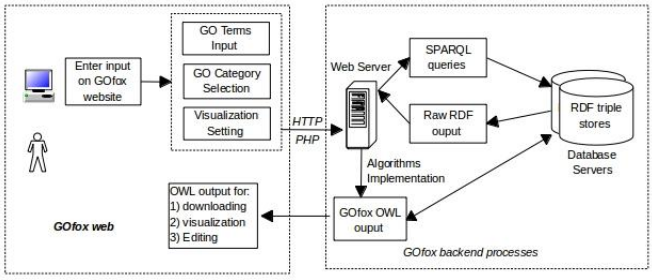


Fig. 1. GOfox program architecture and workflow design.

* To whom correspondence should be addressed: yongqunhe@umich.edu

3 GOFOX NEW ALGORITHM (SIM) FOR PROCESSING ENRICHED GO TERMS

The new GOfox algorithm "Include Computed Simplified Intermediates" (SIM) is developed on the basis of OntoFox "Include Computed Intermediates" (COM). The COM basically removes all intermediate GO terms that match the following rules: 1) the intermediate GO term is not included in the user's input; 2) the intermediate GO term has only one parent and one children GO term (Xiang et al., 2010). Although COM works well for most ontologies, it often does not generate ideal results for ontologies (e.g., GO) that have multiple inheritance. SIM is developed to resolve this issue.

SIM first goes through the COM steps, and the COM results are further simplified by selectively removing some intermediate terms that have multiple parents (e.g., multiple inheritance) based on the following 3 steps. First of all, SIM reformats the OWL-formatted results by removing indirect subclass relationships. For example, the subclass axiom: *(regulates) some (transcription, DNA-templated)* will be removed because the parent-children relationship is not a direct *'is a'* relationship. Second, SIM removes intermediate GO terms that match the following rules: 1) the intermediate GO term is not included in the user's input; 2) if the intermediate GO term has less than two child GO terms within the user's input list (*Note*: here we do not consider one parent condition as COM does). Third, SIM will further trim down the list by removing the subclass relationships between the GO terms and three GO top level terms of BP, CC, and MF. The requirements of the removal are: 1) the term is a direct subclass of BP, CC or MF; 2) there exists another direct subclass relationship between the GO terms and terms other than the three GO top level terms.

While GOfox still keeps the COM algorithm for users to choose, the SIM algorithm provides an extra way of shortening the GO terms in display.

4 GOFOX FEATURES AND WEB INTERFACE

GO provides many features for generating hierarchical classification given a list of user-provided enriched GO terms. Fig. 2 provides a demo on how GOfox works. Specifically, a user can choose to type in GO terms or upload a text file as input. The user can provide a standard P-value or other P-values such as false discovery rate adjusted P-value. A different value cutoff can also be used. The user can then select an intermediates retrieval setting, including COM, SIM, or all intermediates. GOfox will run after "Run GOfox" is clicked (Fig. 2A).

After the results are generated, GOfox provides an Ontobee-like term visualization interface (Fig. 2B). This feature is good for biologists who are not familiar with using the Protégé OWL editor to display output files. The user can interactively explore the hierarchy of retrieved GO terms and also hide unwanted GO terms from the web page.

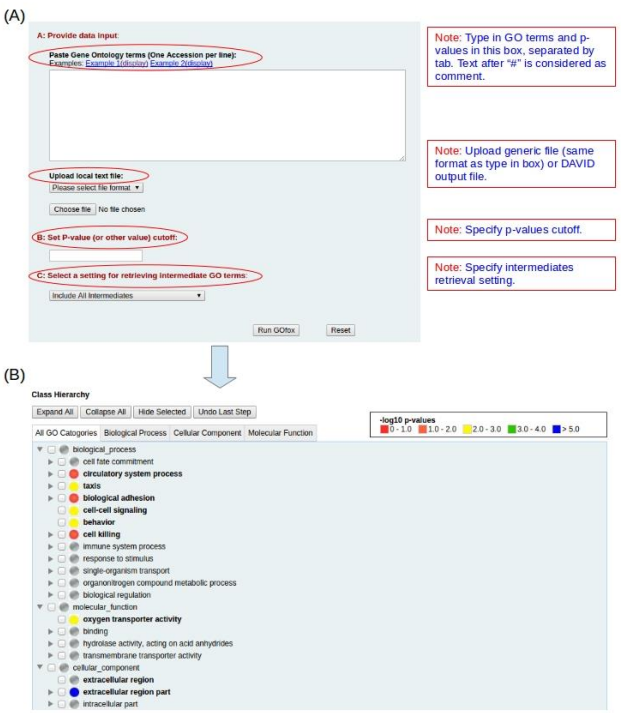


Fig. 2. GOfox website interface and output. (A) GOfox web-interface input form. (B) Standard GOfox SIM algorithm output.

5 AVAILABILITY AND LICENSE

GOfox is freely available on: http://gofox.hegroup.org/. With the license of Apache License 2.0, the source code is released on Github: https://github.com/ontoden/gofox.

6 SUMMARY

GOfox is a simplified hierarchical classification tool to help user interpret the results of GO enrichment analysis. GOfox addresses a critical issue. *i.e.*, the difficulty to visualize, select and further analyze the increased number of enriched GO terms from the popular GO enrichment analysis studies.

REFERENCES

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet* 25**,** 25-29.

Xiang, Z., Courtot, M., Brinkman, R.R., Ruttenberg, A., and He, Y. (2010). OntoFox: web-based support for ontology reuse. *BMC Res Notes* 3:175**,** 1-12.

Xiang, Z., Mungall, C., Ruttenberg, A., and He, Y. (Year). "Ontobee: A linked data server and browser for ontology terms", in: *The 2nd International Conference on Biomedical Ontologies (ICBO)*: CEUR Workshop Proceedings), Pages 279-281 [http://ceur-ws.org/Vol-833/paper248.pdf].

Ontorat: Automatic generation and editing of ontology terms

Yongqun He[1*], Jie Zheng[2], Yu Lin[1]
[1] University of Michigan, Ann Arbor, Michigan, USA; [2] University of Pennsylvania, Philadelphia, PA, USA

ABSTRACT

The web-based Ontorat (http://ontorat.hegroup.org/) program has been developed to automatically generate and edit ontology terms from a spreadsheet format input (Excel or tab-delimited file) based on an ontology design pattern (ODP). The Ontorat web interface is intuitive and suitable for users with basic background in Manchester ontology format scripting, the same language used in the Protégé OWL editor. Ontorat also collects ODPs and provides templates and sample data for future reuse. As a demonstration, Ontorat is applied to automatically generate assay terms with axioms and annotations added into the Ontology for Biomedical Investigations (OBI).

1 INTRODUCTION

Manually developing a new ontology can be very time-consuming. To reduce the time in ontology development, we can first develop an Ontology Design Pattern (ODP) (Noppens and Liebig, 2009), apply the ODP to generate an Excel or text file including ODP-oriented data, and then transform the structured data into an OWL format file. Ontorat is a web-based tool to perform such a task (Xiang et al., 2015).

Ontorat is developed based on the ODP strategy, and its development was inspired by the Quick Term Templates (QTT) procedure generated by the developers of the Ontology for Biomedical Investigations (OBI) (Rocca-Serra et al., 2011). Ontorat is able to convert a QTT template in a spreadsheet into an OWL file. Ontorat also includes other features as described elsewhere in this article.

In this software demo, we will introduce general Ontorat design and workflow, describe features of Ontorat, and provide use case demonstrations to show how Ontorat is used to facilitate new ontology term addition and existing ontology term editing.

2 ONTORAT SOFTWARE DESIGN

From the Ontorat web page, a user can enter setting options and upload the input data file via the Ontorat web input form. The input data file is generated by populating a predesigned template file guided by the ODP as mentioned above. The setting options specify the ontological meanings of the columns in the input data file and axioms between terms. After accepting the input data file and setting options from the user, the web server (via a PHP script) will be able to execute two operations: 1) generation of new ontology classes with logical axioms and annotations, or 2) adding new axioms to existing ontology terms. The Ontorat server will process the user's requests and generate either an Ontorat settings file or an OWL output file. The Ontorat settings file can be stored and reused later (Fig. 1).

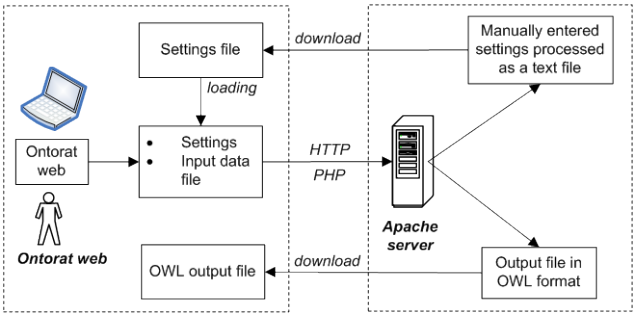


Fig. 1. Ontorat architecture and workflow.

3 ONTORAT WEB INTERFACE

Ontorat includes a user-friendly web form for adding setting options, uploading a data input file, clicking for execution, providing debugging information, and downloading the OWL output.

Input data file: Based on a pre-defined ODP, an Ontorat template file is generated to include all term and annotation types needed for defining a target term. The template file can then be filled up with specific terms and annotations for each type. Ontorat supports both Excel and tab-delimited text input files.

Ontorat settings: Ontology axioms are represented using Manchester OWL Syntax (Horridge et al., 2006). The axiom settings can be added one by one via the Ontorat web form or uploaded from an Ontorat setting text file in an Ontorat-specific setting file format. Ontorat can generate the setting file based on the setting inputs via the Ontorat web form.

Ontorat output file: The OWL output file can be visualized using the Protégé ontology editor (http://protege.stanford.edu/) and imported into the target ontology using the OWL import function.

4 ONTORAT NEW TERM GENERATION

One major function of Ontorat is to generate multiple new ontology terms based on an ODP for a specific ontology.

Fig. 2 provides a demo on how Ontorat can be used to generate a list of assays based on a design pattern (Fig. 2A) out of the Ontology for Biomedical Investigations (OBI) (Brinkman et al., 2010). Fig. 2B provides a portion of the Ontorat setting file. Like the Protégé OWL editor, Ontorat

* To whom correspondence should be addressed: yongqunh@umich.edu

also uses the Manchester OWL Syntax (Horridge et al., 2006) for scripting ontology axioms. For example, Ontorat uses the following Manchester syntax expression:

'has participant' some <http://purl.obolibrary.org/obo/{$columnM}>

to express the relation that a newly generated term has an axiom of 'has participant' some device (which is located in the column M of the Excel input file). Each row in the Excel file (Fig. 2C), starting from row 2, includes the information for generating a new ontology term. The output file can be displayed using Protégé OWL editor (Fig. 2D). The detailed files associated with this use case are available on the Ontorat web page: http://ontorat.hegroup.org/designtemplates/assay/obi-assay.php.

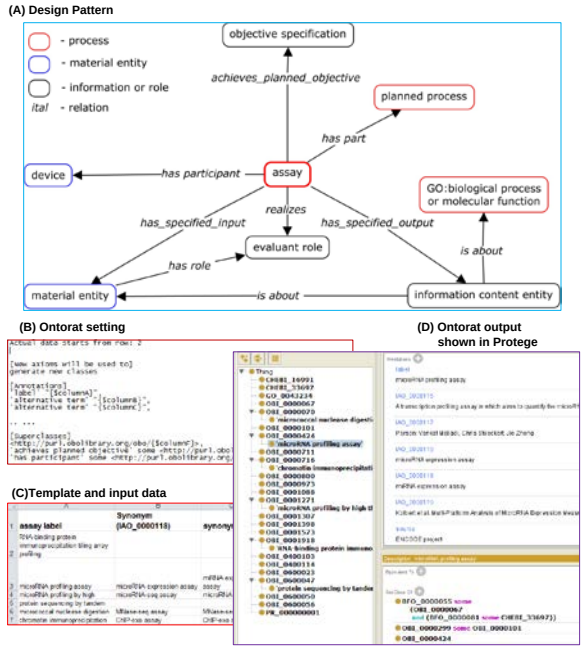


Fig. 2. Ontorat usage for enriching OBI assay terms. (A) Design pattern; (B) Ontorat setting in text format. This text format can be loaded to Ontorat web page directly. (C) Ontorat template together with input data; (D) The output OWL file displayed using the Protégé-OWL editor.

5 ONTORAT EXISTING TERM EDITING

Ontorat also supports editing existing terms by adding new axioms (*e.g.* annotations). For example, Ontorat was used to automatically add definition source and term editor annotations to over 50 ontology classes in the Biobank Ontology (https://code.google.com/p/biobank-ontology/). Such information was not included in the ontology at the early development stage. To add the annotations, the following settings were used in the Ontorat annotations input section:

'definition editor' "{$columnC}",
'definition source' "{$columnD}"

The Ontorat input and output OWL files for this use case are available on the Ontorat website: http://ontorat.hegroup.org/designtemplates/biobank/index.php.

6 COLLECTION OF DESIGN PATTERNS AND TEMPLATES

Ontology design patterns (ODPs) are reusable modeling solutions for ontology development. Ontorat has collected many ODPs and corresponding templates (http://ontorat.hegroup.org/designtemplates.). For each case, Ontorat provides an ODP diagram, an Excel template, a setting file, and an example with populated template data and output OWL file. These ODPs and templates can be reused to support fast and reproducible ontology development.

7 SOURCE CODE AND LICENSE

The Ontorat source code is openly available on GitHub: https://github.com/ontoden/ontorat. The Ontorat source code license is Apache License 2.0.

8 SUMMARY

With ever increasing needs of ontology development and applications, the web-based Ontorat program provides a timely platform for generating and annotating ontology terms based on design patterns.

ACKNOWLEDGEMENTS

This research is supported by a NIH R01 grant (1R01AI081062).

REFERENCES

Brinkman, R.R., Courtot, M., Derom, D., Fostel, J.M., He, Y., Lord, P., Malone, J., Parkinson, H., Peters, B., Rocca-Serra, P., Ruttenberg, A., Sansone, S.A., Soldatova, L.N., Stoeckert, C.J., Jr., Turner, J.A., Zheng, J., and Consortium, O.B.I. (2010). Modeling biomedical experimental processes with OBI. *J Biomed Semantics* 1 Suppl 1**,** S7.

Horridge, M., Drummond, N., Goodwin, J., Rector, A.L., and Wang, H.H. (Year). "The Manchester OWL syntax", in: *OWL: Experiences and Directions (OWLED '06)*, eds. B.C. Grau, P. Hitzler, C. Shankey & E. Wallace: CEUR Workshop Proceedings), 10 pages.

Noppens, Y., and Liebig, T. (Year). "Ontology Patterns and Beyond Towards a Universal Pattern Language", in: *Proceedings of the Workshop on Ontology Patterns (WOP 2009), collocated with the 8th International Semantic Web Conference (ISWC-2009)*).

Rocca-Serra, P., Ruttenberg, A., O'connor, M.J., Whetzel, P.L., Schober, D., Greenbaum, J., Courtot, M., R.R., B., S.A., S., R., S., Consortium, T.O., and Peters, B. (2011). Overcoming the ontology enrichment bottleneck with quick term tem-plates. *Applied Ontology* 6**,** 13-22.

Xiang, Z., Zheng, J., Lin, Y., and He, Y. (2015). Ontorat: Automatic generation of new ontology terms, an-notations, and axioms based on ontology design patterns. *Journal of Biomedical Semantics* 6**,** 4 (10 pages).

OnToology, a tool for collaborative development of ontologies

Ahmad Alobaid [1], Daniel Garijo [1]*, María Poveda-Villalón [1], Idafen Santana-Perez [1] and Oscar Corcho [1]

[1]Ontology Engineering Group, Universidad Politécnica de Madrid, Spain

ABSTRACT

In this demo we present OnToology, a tool for developing ontologies collaboratively using Github. OnToology addresses several steps of the ontology development lifecycle, including documentation, representation, evaluation and publication in a non-intrusive way.

1 INTRODUCTION

The rise of collaborative technologies has sped up the development of software on the last decade. When working as a team, it is common to use repositories for software development, open discussions and having a ticketing system that warns and keeps track of the main issues to be solved.

This paradigm is slowly moving towards other domains, like ontology development. Ontologies, like software, require a set of requirements to be stablished and are usually discussed in a group before agreeing on a design decision. Therefore, they benefit heavily from the ticketing system, versioning and decision tracking that collaborative environments offer. However, this is often not enough, as ontologies need to be further documented and published online. Although some tools cover part of these activities e.g. documentation and evaluation, there are no tools that integrate them with a collaborative environment.

In this demo we present OnToology[1] a tool for documenting, evaluating, presenting and publishing ontologies developed collaboratively. Section 2 describes the requirements for developing ontologies collaboratively, while Section 3 describes our approach. Finally Section 4 describes related work and Section 5 introduces our efforts for improving the tool.

2 ONTOLOGY DEVELOPMENT LIFE CYCLE

Typically, the ontology development process can be divided in several independent activities:

- Ontology requirements: before committing to implement an ontology, it is advised to write a set of competency questions (CQs) in an *ontology requirements specification document* as mentioned in NeOn methodology (Suárez-Figueroa *et al.*, 2012), which will be used to test the ontology.
- Ontology Implementation: once agreed on the ontology requirements, one can use an ontology editor such as NeOn-toolkit[2] or Protégé[3] to design the properties and classes of the proposed ontology.
- Ontology evaluation: the resultant ontology can be evaluated in two different ways: by checking whether the requirements (i.e., CQs) are answered properly and by checking whether the ontology follows design patterns and well stablished practices for its implementation or not.
- Ontology documentation: an ontology is unlikely to be reused unless it is documented properly with examples. This phase focuses in producing a human-readable documentation that allows users understand the OWL or RDFs file produced during the implementation phase.
- Ontology publication: in this phase the ontology has been agreed on and its ready for release. As the aim of the vocabularies and ontologies is normally to share the model for its reuse, the ontology is released with its documentation.

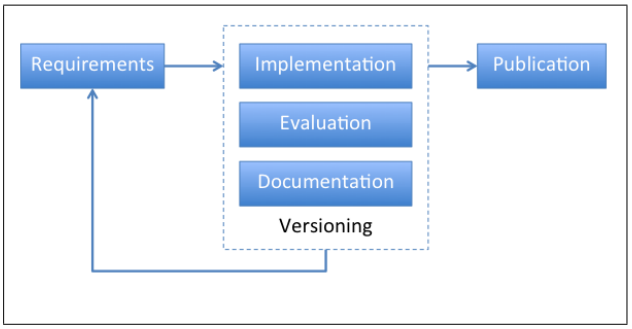


Fig. 1: Ontology development life cycle

Figure 1 presents an overview of the different phases of the ontology development lifecycle. As shown in the figure, this cycle benefits from a collaborative versioning environment that tracks the changes made to the ontology, requirements, documentation and diagrams; and keeps a log of the group discussions and decisions made.

3 COLLABORATIVE CREATION OF ONTOLOGIES WITH ONTOOLOGY

OnToology[4] is a web-based tool designed to automate part of the ontology development process in collaborative environments. In particular, OnToology is designed to work with Github[5], one of the most common environments for software development. After registering a repository to OnToology, developers just push their changes to Github and the tool will produce the documentation (with several proposals for diagram representation), evaluation and publication of the ontology in the user's repository. The phases covered by OnToology are further described below:

*To whom correspondence should be addressed: dgarijo@fi.upm.es

[1] http://purl.org/net/OnToology

[2] http://neon-toolkit.org/

[3] http://protege.stanford.edu/

[4] https://github.com/OnToology/OnToology

[5] http://www.github.com/

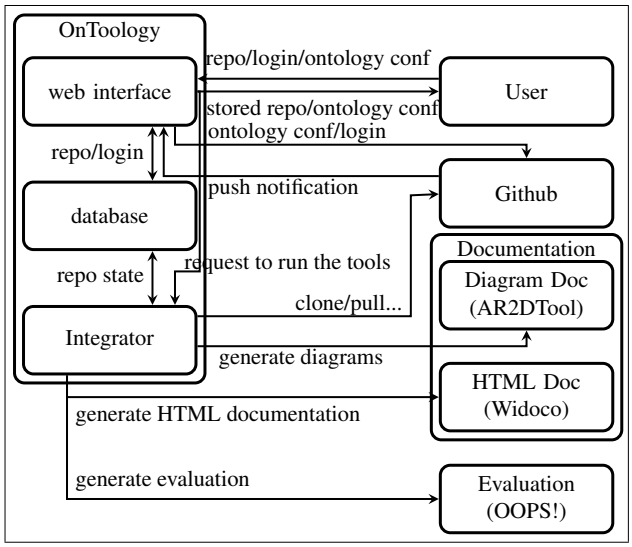


Fig. 2: OnToology Architecture

- **Ontology documentation and depiction:** OnToology integrates AR2DTool[6] for creating taxonomy and entity relationship diagrams of the ontology and Widoco[7], which helps you to create a complete HTML documentation by following a series of steps in a wizard. Widoco is based on LODE (Peroni, S. *et al.*, 2012) and guides the user along the documentation process. It also extracts some of the metadata properties from the ontology, annotates the documentation in RDF-a and creates provenance summary that is W3C PROV-O[8] compliant.
- **Ontology Evaluation:** OnToology integrates OOPS! (OntOlogy Pitfall Scanner!) (Poveda-Villalón *et al.*, 2014), a web based system[9] that helps in the evaluation of OWL ontologies relying mainly on structural and lexical patterns that identify pitfalls in ontologies. OOPS! has been designed to detect up to 33 pitfalls among those defined in the catalogue, and has been used worldwide in different domains. For each ontology, OnToology will create a single issue in Github with a summary of the pitfalls detected by OOPS!, pushing as well an extended explanation to the repository for more information.
- **Ontology Publication:** by using Widoco, OnToology produces a bundle with the documentation that is ready to be deployed on a server.

OnToology allows developers to customize which of the integrated tools are enabled or disabled through a configuration file.

3.1 OnToology Architecture

OnToology is composed of two main parts and is integrated with four external systems as can be seen in Figure 2. The two main parts of the tool are the *web interface* and the *integrator*. The purpose of the *web interface* is to handle Github notifications of the changes of a registered ontology repository and to serve the webpage where users register their repositories after giving the tool access to the repository. There is another webpage served by the *web interface* where users can log into and configure their ontology. The configuration file of an ontology is used by OnToology to enable/disable the generation of each of the tools.

The *integrator* talks to four systems: AR2DTool, Widoco, OOPS! and Github. The first three systems are used by OnToology to produce the diagrams, the HTML documentation and an evaluation report respectively. The integration with Github consists on cloning the repository, adding OnToology user as a collaborator, creating webhooks (that are responsible for notifying OnToology of the changes on repository), aggregating the produced files from the integrated systems and submitting them in a pull request to the repository, where the maintainer can review the changes and merge them later on. The *integrator* also opens an issue in Github including the summary of the pitfalls generated by OOPS! with a link to a full extended explanation.

[6] https://github.com/idafensp/AR2DTool/

[7] https://github.com/dgarijo/Widoco/

[8] http://www.w3.org/TR/2013/REC-prov-o-20130430/

[9] http://oops.linkeddata.es/

4 RELATED WORK

Neologism (Cosmin Basca *et al.*, 2008) is a web-based editor for vocabulary editing and publishing. Unlike OnToology, it lacks revision tracking and ontology best practice evaluation. VoCol (Niklas Petersen *et al.*) is another tool integrated with Github for a collaborative approach. The tool suffers from strictness and lack of freedom on the generated output, while OnToology provides full control over the generated output. Finally, WebProtégé is a web-based ontology editor for the collaborative development of ontologies. WebProtégé is focused on the implementation of the ontologies, which can only be edited online. In OnToology, the creation can be done offline as well. WebProtégé also lacks the evaluation of ontologies like the one provided by OOPS!, which is also used by OnToology.

5 CONCLUSIONS AND FUTURE WORK

In this demo we have introduced OnToology, a tool for improving the ontology development lifecycle in collaborative environments. OnToology helps documenting, depicting, evaluating and publishing. As future work, we are currently working on addressing automatic deployment and archival of the ontology releases under demand.

ACKNOWLEDGEMENTS

This research has been funded by the project “4V: Volumen, Velocidad, Variedad y Validez en la Gestin Innovadora de Datos” (TIN2013-46238-C4-2-R).

REFERENCES

Poveda-Villalón, M., Gómez-Pérez, A., and Suárez-Figueroa, M. C. (2014). Oops!(ontology pitfall scanner!): An on-line tool for ontology evaluation. *International Journal on Semantic Web and Information Systems (IJSWIS)*, **10**(2), 7–34.

Suárez-Figueroa, M.C.; Gómez-Pérez, A; Motta, E.; Gangemi, A. (2012). *The NeOn Methodology for Ontology Engineering*.

Peroni, S., Shotton, D., Vitali, F. (2012). *Easy Vocabulary Publishing. Proceedings of the I-SEMANTICS 2012 Posters & Demonstrations Track, pp. 63-67, 2012*

Cosmin Basca, Stéphane Corlosquet, Richard Cyganiak, Sergio Fernández and Thomas Schandl(2008). *Neologism: Easy Vocabulary Publishing*.

Niklas Petersen, Lavdim Halilaj, Christoph Lange, and Sören Auer. *Neologism: Easy Vocabulary Publishing*.

AberOWL: an ontology portal with OWL EL reasoning

Luke Slater [1]*, Georgios V Gkoutos[2], Paul N Schofield[3], Robert Hoehndorf[1]

[1] Computational Bioscience Research Center, King Abdullah University of Science and Technology, 4700 KAUST, 23955-6900, Thuwal, Saudi Arabia

[2] Department of Computer Science, Aberystwyth University, Aberystwyth, SY23 3DB, Wales, United Kingdom

[3] Department of Physiology, Development and Neuroscience, University of Cambridge, Downing Street, CB2 3EG, England, United Kingdom

ABSTRACT

The field of biological and biomedical science quickly generate large quantities of data and knowledge; often, domain knowledge is formalised using ontologies expressed in the Web Ontology Language (OWL). Ontology repositories such as Bioportal and Ontobee have been an important infrastructural component for managing ontologies, specifically to search, browse and download ontologies over the Web. We present the AberOWL system, a novel ontology repository that allows access to multiple ontologies through automated reasoning, utilizing parts of the OWL of the ontologies alongside a web interface and web services. AberOWL contains over 300 ontologies and integrates reasoning over ontologies with access to literature and SPARQL endpoints.

1 INTRODUCTION

Major ontology repositories such as BioPortal (Noy *et al.*, 2009), OntoBee (Xiang *et al.*, 2011), and the Ontology Lookup Service (Cote *et al.*, 2006), have existed for a number of years, and currently contain several hundred ontologies. They allow ontology creators to upload, manage and release their work to the wider community. For the end-user, they provide a web front-end for browsing, comparing, visualising, downloading, searching, and otherwise processing ontologies.

One feature that the existing ontology repositories lack is to utilize automated reasoning over ontologies to enrich the set of services they provide. Automated reasoning over the axioms in the ontologies enables the use of deductive inference when processing an ontology, which can improve the utility of many services provided by an ontology repository:

- *Verify Consistency and coherence* – The releases and updates of ontologies can be automatically classified by the repository, checking them for logical consistency and existence of unsatisfiable classes, and informing the ontology maintainers and the users of the result.
- *Semantic Query* – Searching can make use of inferred knowledge 'created' during the reasoning process. Furthermore, the results can be integrated in retrieval of text documents or data characterized with ontologies accessible through through public SPARQL endpoints (Jupp *et al.*, 2014; The Uniprot Consortium, 2007; Belleau *et al.*, 2008; Williams *et al.*, 2012).
- *Versioning* – Ontology versions can be compared semantically based on the inferred knowledge.
- *Reasoning API* – Applications building on the platform have access to classified data, and may build applications requiring ontology semantics and inferred knowledge without having to classify the ontology on the client-side.

However, enabling automated reasoning over multiple ontologies is a challenging task since automated reasoning can be highly complex and costly in terms of time and memory consumption (Tobies, 2000). In particular, ontologies formulated in the Web Ontology Language (OWL) (Grau *et al.*, 2008) can utilise statements based on highly expressive description logics (Horrocks *et al.*, 2000), and therefore queries that utilise automated reasoning cannot, in general, be guaranteed to finish in a reasonable amount of time.

Previous approaches to reasoning over a large set of ontologies have often involved working with existing collections of ontologies, usually from one of the large repositories such as Bioportal (Del Vescovo *et al.*, 2011; Sazonau *et al.*, 2013). Several approaches have employed RDFS reasoning (Patel-Schneider *et al.*, 2004) for answering queries over Bioportal's set of ontologies through a SPARQL interface (Salvadores *et al.*, 2012, 2013). However, RDFS semantics is different from the semantics of OWL in which most of the ontologies are formalized. Alternatively, systems such as OntoQuery (Tudose *et al.*, 2013) provide access to ontologies through automated reasoning but limit the number of ontologies.

The AberOWL (Hoehndorf *et al.*, 2015) system is a novel ontology repository which allows access to multiple ontologies through automated reasoning, utilising the OWL semantics of the ontologies. AberOWL mitigates the complexity challenge by using a reasoner which supports only a subset of OWL (i.e., the OWL EL profile (Motik *et al.*, 2009)), ignoring ontology axioms and queries that do not fall within this subset. This enables the provision of polynomial-time reasoning, which is sufficiently fast for many practical uses even when applied to large ontologies.

In the demonstration, we will show AberOWL and its functionality, highlighting the differences between our system over traditional ontology repositories that do not utilize automated reasoning.

2 ABEROWL

AberOWL consists of an ontology repository, web services which facilitate semantic queries over specific ontologies or the entire set of ontologies contained in the repository, and a user interface. It aims to provide a full framework for interacting with ontologies through an automated reasoner.

2.1 AberOWL Server

The core of AberOWL is a server which handles the loading, classification of and interaction with ontologies, exposing its

*To whom correspondence should be addressed: luke.slater@kaust.edu.sa

functionality through a JSON REST API. Specifically, the AberOWL server provides the possibility to query one or all ontologies using the Manchester OWL Syntax. It also supports classifying new ontologies and new versions of existing ontologies, immediately making these available through the API.

Additionally, the AberOWL server provides a searchable index of all classes available in its set of ontologies, including class labels, synonyms, description and other associated information. It utilises an Apache Lucene (Lucene, 2005) index to provide quick searching for classes over the entire set of ontologies. Search over the Lucene index is currently used to facilitate autocompletion in the web interface and provide information about classes in the ontology.

2.2 AberOWL Sync

Another component of AberOWL scans existing ontology repositories, such as BioPortal, and a set of URLs, for new ontologies and new versions of existing ontologies. When a new ontology or a new version of an ontology is found, a request is sent to the AberOWL server to load the new ontology or to retrieve and classify a new version of an existing ontology.

2.3 AberOWL Repository

The AberOWL repository is a web frontend to AberOWL which interacts with the AberOWL server to provide a user interface for querying ontologies, alongside features found in other ontology repositories such as ontology upload, browsing of ontologies, and downloading ontologies and their versions. New ontologies and new versions of ontologies can be uploaded, after which they are immediately classified by the AberOWL server and made available for querying.

The prime goal of the AberOWL repository is to have a user-interface in which all structural information about ontologies is derived in real time from the AberOWL reasoning server. Currently, the interface can be used to upload and classify new ontologies, browse and query ontologies, visualize ontology structure and retrieve information on ontology classes.

3 CONCLUSION AND FUTURE WORK

In the future, we plan to improve AberOWL's potential to retrieve a single class based on search of the class' metadata and axioms. Furthermore, we intend to provide the possibility for the visual comparison of ontology versions, utilizing automated reasoning to show differences in the inferences that can be drawn from both ontologies.

A second area of research is to increase AberOWL's integration with biomedical data. In particular, we aim to more closely integrate AberOWL with publicly available SPARQL endpoints to enable access to these SPARQL endpoints through automated reasoning over ontologies.

ACKNOWLEDGEMENTS

REFERENCES

Belleau, F., Nolin, M., Tourigny, N., Rigault, P., and Morissette, J. (2008). Bio2RDF: Towards a mashup to build bioinformatics knowledge systems. *Journal of Biomedical Informatics*, **41**(5), 706–716.

Cote, R., Jones, P., Apweiler, R., and Hermjakob, H. (2006). The ontology lookup service, a lightweight cross-platform tool for controlled vocabulary queries. *BMC Bioinformatics*, **7**(1), 97+.

Del Vescovo, C., Gessler, D. D., Klinov, P., Parsia, B., Sattler, U., Schneider, T., and Winget, A. (2011). Decomposition and modular structure of bioportal ontologies. In *The Semantic Web–ISWC 2011*, pages 130–145. Springer.

Grau, B., Horrocks, I., Motik, B., Parsia, B., Patelschneider, P., and Sattler, U. (2008). OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web*, **6**(4), 309–322.

Hoehndorf, R., Dumontier, M., Oellrich, A., Wimalaratne, S., Rebholz-Schuhmann, D., Schofield, P., and Gkoutos, G. V. (2011). A common layer of interoperability for biomedical ontologies based on OWL EL. *Bioinformatics*, **27**(7), 1001–1008.

Hoehndorf, R., Slater, L., Schofield, P. N., and Gkoutos, G. V. (2015). Aber-owl: a framework for ontology-based data access in biology. *BMC Bioinformatics*.

Horrocks, I., Sattler, U., and Tobies, S. (2000). Practical reasoning for very expressive description logics. *Logic Journal of the IGPL*, **8**(3), 239–264.

Jupp, S., Malone, J., Bolleman, J., Brandizi, M., Davies, M., Garcia, L., Gaulton, A., Gehant, S., Laibe, C., Redaschi, N., Wimalaratne, S. M., Martin, M., Le Novre, N., Parkinson, H., Birney, E., and Jenkinson, A. M. (2014). The EBI RDF platform: linked open data for the life sciences. *Bioinformatics*, **30**(9), 1338–1339.

Lucene, A. (2005). A high-performance, full-featured text search engine library. *URL: http://lucene. apache. org*.

Manola, F. and Miller, E., editors (2004). *RDF Primer*. W3C Recommendation. World Wide Web Consortium.

Motik, B., Grau, B. C., Horrocks, I., Wu, Z., Fokoue, A., and Lutz, C. (2009). Owl 2 web ontology language: Profiles. Recommendation, World Wide Web Consortium (W3C).

Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A. A., Chute, C. G., and Musen, M. A. (2009). Bioportal: ontologies and integrated data resources at the click of a mouse. *Nucleic acids research*, **37**(Web Server issue), W170–173.

Patel-Schneider, P. F., Hayes, P., and Horrocks, I. (2004). Owl web ontology language semantics and abstract syntax section 5. rdf-compatible model-theoretic semantics. Technical report, W3C.

Salvadores, M., Horridge, M., Alexander, P. R., Fergerson, R. W., Musen, M. A., and Noy, N. F. (2012). Using sparql to query bioportal ontologies and metadata. In *The Semantic Web–ISWC 2012*, pages 180–195. Springer.

Salvadores, M., Alexander, P. R., Musen, M. A., and Noy, N. F. (2013). Bioportal as a dataset of linked biomedical ontologies and terminologies in rdf. *Semantic web*, **4**(3), 277–284.

Sazonau, V., Sattler, U., and Brown, G. (2013). Predicting performance of owl reasoners: Locally or globally? Technical report, Technical report, School of Computer Science, University of Manchester.

Seaborne, A. and Prud'hommeaux, E. (2008). SPARQL query language for RDF. W3C recommendation, W3C. http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.

The Uniprot Consortium (2007). The universal protein resource (uniprot). *Nucleic Acids Res*, **35**(Database issue).

Tobies, S. (2000). The complexity of reasoning with cardinality restrictions and nominals in expressive description logics. *J. Artif. Int. Res.*, **12**(1), 199–217.

Tudose, I., Hastings, J., Muthukrishnan, V., Owen, G., Turner, S., Dekker, A., Kale, N., Ennis, M., and Steinbeck, C. (2013). Ontoquery: easy-to-use web-based owl querying. *Bioinformatics*, **29**(22), 2955–2957.

Williams, A. J., Harland, L., Groth, P., Pettifer, S., Chichester, C., Willighagen, E. L., Evelo, C. T., Blomberg, N., Ecker, G., Goble, C., and Mons, B. (2012). Open phacts: semantic interoperability for drug discovery. *Drug Discovery Today*, **17**(2122), 1188 – 1198.

Xiang, Z., Mungall, C. J., Ruttenberg, A., and He, Y. (2011). Ontobee: A linked data server and browser for ontology terms. In *Proceedings of International Conference on Biomedical Ontology*, pages 279–281.

EDN-LD: A simple linked data tool

James A. Overton[1]*
[1]Knocean, Toronto, Ontario, Canada

ABSTRACT

EDN-LD is a set of conventions for representing linked data using Extensible Data Notation (EDN) and a library for conveniently working with those representations in the Clojure programming language. It provides a lightweight alternative to existing linked data tools for many common use cases, much in the spirit of JSON-LD. We present the motivation and design of EDN-LD, and demonstrate how it can clearly and concisely transform tables into triples.

1 INTRODUCTION

EDN-LD is a set of conventions for representing linked data using Extensible Data Notation (EDN),[1] and a library for conveniently working with those representations in the Clojure programming language.[2] Clojure is a modern Lisp that runs on the Java Virtual Machine (JVM) and has full access to the vast ecosystem of Java libraries. Since many linked data libraries and tools also target the JVM, Clojure is a tempting alternative to Java for working with linked data. Tawny-OWL is another example of a linked data tool written in Clojure (Lord, 2013), however it is focused on ontology development and takes quite a different approach from EDN-LD. With this project our goal is to provide a lightweight alternative to existing linked data tools for many common use cases, much in the spirit of JSON-LD.[3] In this presentation we discuss the motivation and design of EDN-LD, and demonstrate how it can clearly and concisely transform tables into triples.

EDN-LD is open source software, published under a BSD license. The source code is written in a literate style, with extensive unit tests. It is available on GitHub[4] with a tutorial that also serves as an automated integration test. Our interactive online tutorial can be used without needing to install Clojure.[5] Feedback and contributions are welcome on our GitHub site.

2 JSON-LD

EDN-LD shares many of the motivations and goals of JSON-LD. First we will discuss the benefits and shortcomings of JSON-LD, then show how EDN-LD improves on it in several respects.

JavaScript Object Notation (JSON)[6] is a subset of the JavaScript programming language that is widely used for expressing literal data within JavaScript programs. JSON's elements are: null, booleans, strings, and numbers. Elements can be combined into arrays and objects, where the latter are effectively maps from strings to other values. These simple elements are common to virtually every programming language, and JSON is now widely used for data transfer between programs. It has replaced heavier formats such as XML in many applications.

JSON has many limitations, including a lack of comments, ambiguous numbers, and the lack of any mechanism for extending its types. In practise, strings are used to represent most types of data, but since it is difficult to attach type information to aid in their interpretation, this can quickly lead to ambiguity.

The ubiquity of JSON was one motivation for the JSON-LD W3C Recommendation: "A JSON-based Serialization for Linked Data".[7] In JSON-LD strings are used to represent IRIs (and compact IRIs) for resources, plain literals can be strings, and typed literals are objects (maps) with a special `@value`, `@type`, and `@language` keys. Graphs and datasets are represented as nested objects (maps) and sets are represented by arrays, with details depending on the chosen "Document Form".

The core of JSON-LD is the `@context` map, which can be specified inside a JSON record, externally using a link, or provided by the consuming application. The context allows for strings to be interpreted as IRIs, for compact IRI strings to be expanded, and for types to be attached to literals. Since the context can be supplied externally, existing JSON data can be reinterpreted as JSON-LD by providing an appropriate context.

JSON-LD is an exciting addition to the ecosystem of linked data tools, but it is constrained by the limitations of the JSON format. The heavy use of strings, in particular, can make it difficult to distinguish between a literal string, a compact IRI, or a fully resolved IRI. The complex context processing[8] and expansion algorithms[9] are indicative of this problem, as is the need for several similar-but-different "Document Forms". EDN-LD uses the richer elements and structures available in EDN to reduce these problems.

3 EXTENSIBLE DATA NOTATION

Like JSON and JavaScript, Extensible Data Notation (EDN) is the a data format at the core of Clojure. The basic EDN elements are: nil, booleans, strings, characters, symbols, keywords, integers, and floating point numbers. These can be combined into lists, vectors, maps, and sets. Any element can serve as the key or value of a map. EDN is extensible in the sense that it allows for *tagged elements*, indicated by a special tag followed by an EDN element. EDN also allows two kinds of comments. Multiple alternatives to strings (i.e. keywords and symbols), more carefully defined numbers, sets, and more flexible maps all make it easier to express complex data efficiently and unambiguously in EDN than in JSON.

*To whom correspondence should be addressed: james@overton.ca

[1] https://github.com/edn-format/edn

[2] http://clojure.org

[3] http://json-ld.org

[4] https://github.com/ontodev/edn-ld

[5] http://try.edn-ld.com

[6] http://json.org

[7] http://www.w3.org/TR/json-ld/

[8] http://www.w3.org/TR/json-ld-api/#context-processing-algorithms

[9] http://www.w3.org/TR/json-ld-api/#expansion-algorithms

EDN does not have a type system and does not include schemas. However several schema systems have been created for validating EDN data structures. EDN-LD uses Prismatic's Schema library[10] to specify the required "shapes" for various elements.

4 EDN FOR LINKED DATA

In EDN-LD as in JSON-LD, IRIs and blank node identifiers are represented by strings. IRIs can be *contracted* to keywords using a context: a map from keywords to IRIs or other contractions. Contractions can be expanded to IRIs using the same context. Literals are always represented as maps with a special `:value` key for the lexical value, and optional `:type` and `:lang` keys. Discrete triples and quads are represented with vectors. Graphs and datasets are represented as nested maps from graph IRI to subject IRI to predicate IRI, ending with a set of objects. These two "document forms" have very different shapes, suited to different processing goals, e.g. sequences of triples for streaming and filtering, and nested maps for sorting and selecting. EDN-LD uses Apache Jena[11] to read from and write to a wide range of linked data formats.

Figure 1 shows an EDN-LD context. It includes a `:dc` prefix for Dublin Core metadata elements, and an `:ex` prefix for the example domain. The `nil` key indicates that its value `:ex` is the default prefix. The `:title` and `:author` contractions expand (recursively) to Dublin Core IRIs.

```
(def context
  {:dc     "http://purl.org/dc/elements/1.1/"
   :ex     "http://example.com/"
   nil     :ex
   :title  :dc:title
   :author :dc:author})
(expand context :title)
; "http://purl.org/dc/elements/1.1/title"
```

Fig. 1. An example of an EDN-LD context, showing an `expand` function call on the `:title` contraction, and the expanded IRI that is returned

Figure 2 shows an example of a simple data conversion pipeline using EDN-LD. First we define a map from names (strings) to contracted resource IRIs and merge our context with the default prefixes. The `->>` is a "threading macro" that inserts the first value as the last argument to the second function, and so on, letting deeply nested function calls be clearly expressed as "pipelines". Here "books.tsv" is the name of a file in tab-separated values format and `read-tsv` is a function that returns a sequence of maps for each row, each with column names as keys. The `assign-subject-iri` function is called on each of the maps to add a `:subject-iri` key with appropriate value. Then `triplify` is used to convert the maps to triples, represented as vectors: subject keyword, predicate keyword, and object keyword or literal map as determined by the `resources` map. The keywords represent contracted IRIs, and the `expand-all` function converts them to full IRI strings. Finally the `write-triples` function writes the results to the "books.ttl" Turtle file using our specified prefixes. By using keywords to distinguish contracted IRIs from full IRIs and literal data, and consistently using maps for literal data, we gain more control over the interpretation of strings than JSON-LD, without loss of concision.

```
(def resources
  {"Homer" :Homer})
(def prefixes
  (merge
    default-prefixes
    context))
(->> "books.tsv"
     read-tsv
     (map assign-subject-iri)
     (mapcat #(triplify resources %))
     (map #(expand-all context %))
     (write-triples "books.tsv"
                    prefixes))
```

Fig. 2. An example of an EDN-LD conversion pipeline

5 FUTURE WORK

EDN-LD is still in development, but available for use. We plan to implement convenient syntax for RDF collections (linked lists), and for various OWL constructs including annotation axioms and class expressions. We are also considering a ClojureScript implementation of EDN-LD. ClojureScript is a language that is closely related to Clojure, compiling to JavaScript rather than JVM bytecode. Dual Clojure and ClojureScript libraries are becoming increasingly common. But a ClojureScript version of EDN-LD could not use Jena, and would need alternative methods for reading and writing linked data files.

6 DEMONSTRATION

At ICBO we plan to demonstrate the use of EDN-LD for transforming tables to triples, and for efficiently filtering large linked data files to specified subsets.

7 CONCLUSION

EDN-LD was developed for the Immune Epitope Database (IEDB), and was preceded by several related systems for working with linked data and ontologies using Clojure. These techniques have proved valuable for rapid development of data processing workflows, merging disparate sources of biological data. EDN-LD improves on JSON-LD in several respects, and is well suited to working with linked data in Clojure.

ACKNOWLEDGEMENTS

The author was supported in this work by the Immune Epitope Database and Analysis Project, funded by the National Institutes of Health [HHSN272201200010C].

REFERENCES

Lord, P. (2013). The Semantic Web takes wing: Programming ontologies with Tawny-OWL. *OWLED 2013*.

[10] `https://github.com/Prismatic/schema`

[11] `http://jena.apache.org`

ROBOT: A command-line tool for ontology development

James A. Overton,[1]* Heiko Dietze,[2] Shahim Essaid,[3]
David Osumi-Sutherland,[4] and Christopher J. Mungall [2]

[1]Knocean, Toronto, Ontario, Canada
[2]Lawrence Berkeley National Laboratory, Berkeley, California, USA
[3]Oregon Health and Science University Library, Portland, Oregon, USA
[4]European Molecular Biology Laboratory – European Bioinformatics Institute,
Wellcome Trust Genome Campus, Hinxton, United Kingdom

ABSTRACT

ROBOT is a command-line tool for working with ontologies, especially Open Biomedical Ontologies. It builds on OWLAPI and is designed to eventually replace Oort and many functions of OWLTools. Currently implemented commands include: reporting on differences between ontologies, merging ontologies, extracting ontology modules, filtering ObjectProperties, and reasoning. Commands can be chained together to form powerful, repeatable workflows. ROBOT is in early development but is available for use under an open source (BSD) license.

1 INTRODUCTION

ROBOT is a new command-line tool for working with ontologies, with special emphasis on Open Biomedical Ontologies (OBO) (Smith *et al.*, 2007). It provides convenient commands for merging ontologies, extracting subsets, filtering for selected axioms, running reasoners, and converting between file formats. Commands can be chained together to form powerful, repeatable workflows.

OWLTools and Oort are the predecessors to ROBOT. OWLTools provides various functionality to support the Gene Ontology (GO) and many other ontology projects. In particular, it has supported the transition of GO and other projects from OBO format toward OWL format (Mungall *et al.*, 2014). Oort, the OBO Ontology Release Tool, is part of OWLTools and specifically designed as a command-line tool to help automate the transformation of ontology files from editing versions to publishable release files. Both are used extensively by BerkeleyBOP and other projects, often scripted by GNU Make files and run as part of Continuous Integration systems (Mungall *et al.*, 2012).

OWLTools and Oort provide more core functionality than ROBOT currently does. However, they have "evolved" over the years without an overarching design. The resulting tools are powerful, but somewhat difficult for new users to learn, and the code base was designed around certain assumptions that held true for OBO format ontologies but do not apply to OWL format ontologies.

With ROBOT we aim to create a more modular and extensible code base for developers, and provide a friendlier and more consistent interface for users. We plan to incrementally replace OWLTools and Oort, and promote the use of ROBOT as a standard tool for use by OBO projects. In this paper we describe the design of ROBOT, its usage, and some of our future plans.

*To whom correspondence should be addressed: james@overton.ca

ROBOT is Open Source software, released under a BSD license, and is available on GitHub.[1] Although ROBOT is in early development, it is available for download as a JAR file and with executable scripts for Unix (including Mac OS X and Linux) and Windows platforms. We appreciate feedback, especially in the form of issues and pull-requests on GitHub.

2 OPERATIONS

ROBOT is a Java project built with Apache Maven[2], and divided into two modules:

- `robot-core` implements the basic operations
- `robot-command` implements the command-line interface

`robot-core` is designed to be used as a library from any language that runs on the Java Virtual Machine. Operations are classes that provide static methods for working with ontologies, usually building on functionality from OWLAPI (Horridge and Bechhofer, 2011). The currently implemented operations are:

- `DiffOperation`: find the differences between two ontologies
- `ExtractOperation`: extract a module
- `FilterOperation`: filter axioms by ObjectProperty
- `MergeOperation`: merge axioms from one or more ontologies and their imports into a single ontology
- `ReasonOperation`: use a reasoner to add axioms to an ontology

We plan to provide operations for MIREOT (Courtot *et al.*, 2011), for Quick Term Templates (Rocca-Serra *et al.*, 2011), for validating ontologies using SPARQL queries, and for modifying ontologies using SPARQL Update. `robot-core` also provides a class with static utility methods and a class for convenient file operations.

3 COMMANDS AND CHAINS

The `robot-command` module implements ROBOT's command-line interface using the Apache Commons CLI library.[3]. Each command implements the `Command` interface, with `main` and `execute` methods, as well as other common methods for getting the command name and usage information. Commands do not necessarily correspond to operations, but often do. The currently implemented commands are:

[1] `https://github.com/ontodev/robot`

[2] `https://maven.apache.org`

[3] `https://commons.apache.org/proper/commons-cli/`

- `AnnotateCommand`: add annotations to an ontology
- `ConvertCommand`: save an ontology to another format: RDFXML, RDFOWL, Turtle, Manchester Syntax, OWL Functional Syntax, or OBO.
- `DiffCommand`: show the differences between two ontologies
- `ExportPrefixesCommand`: show the current prefixes (see below)
- `ExtractCommand`
- `FilterCommand`
- `MergeCommand`
- `ReasonCommand`

Commands can be called individually or by `CommandManager` instances. When used with a manager, commands can be *chained* together into powerful workflows. When chained, commands communicate using a `CommandState` object that contains the current `OWLOntology` instance. Internally, command accepts a state, updates it, and returns it for the next command to use. At the command-line, each command is named by a verb and followed by zero or more options and their arguments. Figure 1 shows an example of a chain of commands for releasing an ontology.

```
robot \
  merge --input edit.owl \
  reason --reasoner ELK \
  annotate \
    --annotation-file annotations.ttl \
    --output results/example.owl \
  convert --output results/example.obo
```

Fig. 1. An example of a chain of commands

4 NAMES AND FORMATS

One source of problems with OWLTools has been the use of prefixed identifiers in OBO and OWL. With ROBOT we have made use of the JSON-LD standard[4] and the JSONLD-Java[5] library to provide a concise and effective method for specifying and resolving prefixes. A JSON-LD *context* is a JavaScript object with keys and values that specify how names and prefixes should be expanded, optionally with additional type information. The JSON-LD specification provides an algorithm for resolving names and prefixes to full IRIs.

ROBOT supports the use of JSON-LD for specifying prefixes to be used when loading and saving ontologies and sets of ontology terms. We also support loading ontology and annotation data from JSON-LD files, and from YAML files[6] when they have certain structure corresponding to JSON-LD.

5 DOCUMENTATION AND TESTING

With ROBOT we aim to provide a user-friendly tool for ontology users and ontology developers. The project repository includes a README with installation instructions, a tutorial on command-line usage with example data files, and full JavaDoc documentation for the code itself. The command-line tool provides interactive help on commands. We plan to include ROBOT in the online OBO Tutorial,[7] replacing custom Java code with standard ROBOT commands.

In addition to a suite of unit tests, the tutorial itself serves as an integration test suite. A test harness extracts the example commands from the tutorial document, runs them, and compares the resulting files to the known-good example data files. Continuous Integration systems run all tests every time code is committed to version control.

6 DEMONSTRATION

At ICBO we plan to demonstrate the installation and use of ROBOT for a range of ontology operations, based on our tutorial document. ROBOT will also be used in the OBO Tutorial session at the conference.

7 CONCLUSION

ROBOT is a user-friendly tool for working with ontologies at the command-line and for scripting ontology workflows. It is designed to replace Oort and many of the functions of OWLTools, and to address a wider audience of ontology developers. While still in early development, ROBOT can be used today, and we appreciate feedback via our GitHub repository.

ACKNOWLEDGEMENTS

JAO, HD, DOS, and CJM were supported by the National Human Genome Research Institute (NHGRI) P41 grant 5P41HG002273-09 to the Gene Ontology Consortium. In addition, HD and CJMs contribution was also supported by the Director, Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under Contract No. DE- AC02-05CH11231. JAO's contribution was also supported by the Immune Epitope Database and Analysis Project, funded by the National Institutes of Health [HHSN272201200010C].

REFERENCES

Courtot, M., Gibson, F., Lister, A. L., Malone, J., Schober, D., Brinkman, R. R., and Ruttenberg, A. (2011). MIREOT: The minimum information to reference an external ontology term. *Applied Ontology*, **6**(1), 23–33.

Horridge, M. and Bechhofer, S. (2011). The OWL API: A Java API for OWL ontologies. *Semantic Web*, **2**(1), 11–21.

Mungall, C. J., Dietze, H., Carbon, S., Ireland, A., Bauer, S., and Lewis, S. (2012). Continuous Integration of Open Biological Ontology libraries. *Bio-Ontologies*.

Mungall, C. J., Dietze, H., and Osumi-Sutherland, D. (2014). Use of OWL within the Gene Ontology. *bioRxiv*.

Rocca-Serra, P., Ruttenberg, A., O'Connor, M. J., Whetzel, P. L., Schober, D., Greenbaum, J., Courtot, M., Brinkman, R. R., Sansone, S. A., Scheuermann, R., *et al.* (2011). Overcoming the ontology enrichment bottleneck with quick term templates. *Applied Ontology*, **6**(1), 13–22.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., The OBI Consortium, Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Scheuermann, R. H., Shah, N., Whetzel, P. L., and Lewis, S. (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotechnol*, **25**(11), 1251–1255.

[4] `http://www.w3.org/TR/json-ld/`

[5] `https://github.com/jsonld-java/jsonld-java`

[6] `http://yaml.org`

[7] `https://github.com/jamesaoverton/obo-tutorial`

Highly Literate Ontologies

Phillip Lord* and Jennifer D. Warrender

School of Computing Science, Newcastle University, Newcastle-upon-Tyne, UK

ABSTRACT

There is still a lot of discussion about exactly what ontologies should represent, but what is generally agreed is that they formalise and relate to some relatively complex areas of knowledge. While ontology environments allow rich descriptions of the relationship between the entities inside the ontology (because this is what an ontology is), they often do not provide the same rich environment to describe the knowledge that they represent.

OWL does, for instance, supports annotations which allows an ontology developer to add comments to many parts of the ontology. But these comments, do not contain markup, sectioning or any of the standard facilities authors use when writing documents.

Our solution to this builds on Tawny-OWL, our highly-programmatic environment for ontology development. This provides a rich environment, which allows abstraction, automation and extension, while still being entirely textual. As a result, it is possible to integrate this form of ontology with similar textual environments for documentation such as LaTeX, or AsciiDoc. We call the result a literate ontology, in reference to literate programming. The result can be "tangled" to produce either a document or ontology.

However, manipulating mixed syntax formats is difficult. Generally, the text editor either supports the literate form or programmatic (ontology) form best. To address this, we have developed what we call "lenticular views" – essentially, the source code can be presented either in an ontology-centric or a document-centric view. Either form can be changed, giving the author a powerful and unique environment for creating literate ontologies. Or alternatively, semantic documents where the ontology formalises the document. We demonstrate this with our literate amino-acid ontology which is also a part of the developing manual for Tawny-OWL.

1 INTRODUCTION

Ontologies have been used extensively to describe many parts of biology. Ontologies have two key features which make their usage attractive. First, they provide a mechanism for standardising and sharing the terms used in descriptions, making comparison easier and, secondly, they provide a computationally amenable semantics to these descriptions, making it possible to draw conclusions about the relationships between descriptions even when they share no terms in common.

Despite these advantages, the oldest and most common form of description in biology is free text, or a semi-structured representation through the use of a standardised fill-in form. Free text has numerous advantages compared to ontologies: it is richly expressive, is widely supported by tooling, and while the form of language used in science ("Bad English" (Wood *et al.*, 2001)) may not be easy to use, understand or learn, it is widely taught and most scientists are familiar with it.

*To whom correspondence should be addressed: phillip.lord@newcastle.ac.uk

The two forms of description have largely been used independently. Ontology terms are sometimes used in semi-structured formats such as a UniProt record, or minimum information documents. While these use ontologies in some parts of the document, in general, ontology terms and the free text are in different parts of the record. In this paper, we show how can we integrate ontological and textual knowledge in a single authoring environment and describe how we are applying this to describing amino acids.

2 DEVELOPING KNOWLEDGE

First, we ask the question, why is it difficult to relate ontological and textual descriptions. One possible explanation is that the two forms have very different "development environments". The main documentation environment used within science is Word, followed by LaTeX, common in more mathematical environments. More recently, there has also been interest in various light-weight markup languages, such as markdown, and their associated tool-chains.

Ontology development environments also come in many different forms. Early versions of the Gene Ontology, for instance, used a bespoke text file format and a text editor – an approach rather similar to the light-weight markup languages of today. This had the significant advantage of a low-technological barrier to entry. More modern environments provide a much more graphical interface. These generally provide a much richer way of interacting with an ontology.

While these environments add a lot of value, they do not necessarily integrate well with text. Both Protégé and OBO-Edit have a class-centric view and are biased toward showing the various logical entities in the ontology, as opposed to the textual aspects. Indeed, this bias is shown even at the level of OWL. For example, annotations on an entity (or rather an axiom) are a *set* rather than a list, while ordering is generally considered to be essential for most documents.

With this divergence of development environments, it seems hard to understand how we could square the circle of combining text and ontology development. Next, we describe the amino-acid ontology and how the novel development methodology we used for this ontology allows us to achieve this.

3 TAWNY-OWL

Tawny-OWL (Lord, 2013) provides a fully programmatic environment for development. Simple ontological statements can be written with a syntax inspired by Manchester OWL notation (Horridge and Patel-Schneider, 2012); repetitive statements can be built automatically by writing functions which encapsulate and abstract over the simpler statements, a process we call "patternisation" (Warrender and Lord, 2013).

In this way, we have managed to combine the advantages of text-based environments for editing ontologies i.e. the use of a standard

```
First, to explain the domain. Proteins are polymers
made up from amino-acid monomers. They consist of a
central carbon atom, attached to a carboxyl group (the
``acid'' amino) and amine group (the ``amino'' group)
a hydrogen and an R group. The R group defines the
different amino acids. The different R groups have
different phyiscal or chemical properties, such as
their degree of hydrophobicity. We call these different
characteristics |RefiningFeatures|.

\begin{tawny}
(defclass AminoAcid)

(defclass RefiningFeature)
(defclass PhysicoChemicalProperty :super RefiningFeature)
\end{tawny}
```

Fig. 1. The document-centric view

```
;; First, to explain the domain. Proteins are polymers
;; made up from amino-acid monomers. They consist of a
;; central carbon atom, attached to a carboxyl group (the
;; ``acid'' amino) and amine group (the ``amino'' group)
;; a hydrogen and an R group. The R group defines the
;; different amino acids. The different R groups have
;; different phyiscal or chemical properties, such as
;; their degree of hydrophobicity. We call these different
;; characteristics |RefiningFeatures|.

;; \begin{tawny}
(defclass AminoAcid)

(defclass RefiningFeature)
(defclass PhysicoChemicalProperty :super RefiningFeature)
;; \end{tawny}
```

Fig. 2. The ontology-centric view

editing environment and integration with version control, while maintaining (and in some ways surpassing) the power of tools like Protégé.

Tawny-OWL can be used to generate any ontology, but we demonstrate it here with the amino-acid ontology: a highly patternised ontology with over 430 classes generated from one pattern. We consider next the implications that this has for the ability to integrate ontological and textual descriptions.

4 LITERATE ONTOLOGY

As Tawny-OWL is based on a full programming language, it supports a feature which at first seems quite inconsequential: comments. As with almost every programming language, it is possible to add free, unstructured text to the same source code that defines the ontology. While opinions vary on the role of comments in programmatic code, perhaps the most extreme is that of literate programming (Knuth, 1984) which suggests that code should be usable both as a program capable of execution and as a document capable of reading and that neither view should have primacy.

Literate programming can be difficult, however, partly because the editing environment offers few facilities for it: fundamentally, supporting mixed-syntax text in a tool is a difficult task. Our solution uses a multi-view approach to editing, which allows the author to see her source code in either a *document-centric* or an *ontology-centric* view. We call this approach *lenticular* text, named after lenticular printing which produces images which change depending on your angle of viewing. This is an entirely novel solution to literate programming as it effectively performs the tangling operation for the author as they type. A representation of the two views are shown in Figures 2 and 4. The two views, it should be noted, contain the same **text** but are syntactically different, such that the document-centric view is entirely valid LaTeX code, while the ontology-centric view is valid Tawny-OWL code.

We have now implemented lenticular text for the editor, Emacs[1], in a package called "lentic"[2]. A key feature of this implementation is that both views exist simultaneously in Emacs, and provide all the features of the appropriate development environment; for example, "tab-completion" works in both the document-centric view (completing LaTeX macros) and in the ontology-centric view (completing ontology identifiers). We can launch a compilation of the document-centric view (producing a PDF), or evaluate our ontology, perhaps reasoning over it, in the code-centric view. Therefore, we have achieved a key aim of literate programming: neither view holds primacy and the author can edit either.

5 DISCUSSION

In this paper, we have described our methodology for integration of text and ontological statements at authoring time, using lenticular text to enable literate ontology development. Indeed, we have fully documented the whole of the amino-acid ontology into literate form[3].

The combination of Tawny-OWL and lenticular text is an extremely rich environment. We are aware, however, that it is a specialist environment. To make full use of Tawny-OWL, the author needs to use a Clojure based-development environment, document authoring in LaTeX, and the lentic package which is Emacs-based. In reality, though, the tools are not tightly coupled: we have alternatives beyond LaTeX, Emacs, or even Tawny-OWL. At the same time, one output form of a literate ontology is a readable PDF document, something far more familiar to biologists or medics than Protégé or any ontology development environment.

ACKNOWLEDGEMENTS

This work was supported by Newcastle University.

REFERENCES

Horridge, M. and Patel-Schneider, P. F. (2012). Owl 2 web ontology language manchester syntax (second edition). Technical report.

Knuth, D. E. (1984). Literate programming. *The Computer Journal*, **27**, 97–111.

Lord, P. (2013). The Semantic Web takes Wing: Programming Ontologies with Tawny-OWL. *OWLED 2013*.

Warrender, J. D. and Lord, P. (2013). A pattern-driven approach to biomedical ontology engineering. *SWAT4LS 2013*.

Wood, A., Flowerdew, J., and Peacock, M. (2001). International scientific english: The language of research scientists around the world. *Research Perspectives on English for Academic Purposes*, pages 71–83.

[1] https://www.gnu.org/software/emacs/

[2] https://github.com/phillord/lentic

[3] https://github.com/phillord/tawny-tutorial

NCBO BioPortal Version 4

Ray W. Fergerson, Paul R. Alexander, Rafael S. Gonçalves, Manuel Salvadores, Alex Skrenchuk, Jennifer Vendetti, and Mark A. Musen
Stanford Center for Biomedical Informatics Research
Stanford University, USA
rafael.goncalves@stanford.edu

ABSTRACT

The BioPortal web application of the National Center for Biomedical Ontology is a resource that provides access to more than 600 biomedical ontologies. BioPortal users can browse ontologies with any web browser and search for terms across all the ontologies. Users may also access all ontology content and metadata through a Web services interface; many users have taken advantage of this feature to build applications on top of BioPortal. We present here the most recent updates to the system, featured in version 4 of the BioPortal software – roughly corresponding to the past year, as well as planned extensions to BioPortal.

NEW FEATURES AND UPDATES

In the version 4 release of BioPortal, we have introduced a series of changes and new features to make it easier both to use the web interface to explore ontologies, and to access the system through web services. We summarize these changes below.

Search: The search system allows a user to search for a class in all of the ontologies in BioPortal. We have revised this system to fetch matches not only according to class names and synonyms, but also on other fields such as UMLS CUI and ID. Additionally, we have reworked the search results display to highlight the distinction between ontologies that define a class and those that simply reuse or import a class.

Annotation: The annotator allows a user to submit text to the system and have the system return classes that match concepts in the submitted text. We have revised the annotation system so that new ontologies, and revisions of existing ontologies, are available for annotation within hours of their being loaded into the system. Previously this process took weeks or months.

Mappings: The mappings system records relationships between classes in different ontologies. These relationships can be generated manually (by users) or automatically (by the system). We have revised the mappings system to significantly reduce the resources required, and to substantially speed up their retrieval. Moreover, our collaborators at the University of Victoria have generated a new system for visualizing mappings between sets of ontologies.

Resource Index: The Resource Index is a pre-compiled set of annotations for all ontologies in BioPortal over the contents of a publicly available resource (such as PubMed abstracts). We have re-engineered the Resource Index to improve its scalability, now allowing us to index all years of PubMed and about 50 other biomedical resources.

Ontology Recommender: The Ontology Recommender is a system that helps users identifying ontologies of interest in a particular domain; it accepts text and returns ontologies that are most relevant for annotating this text. We have re-engineered this tool to provide better recommendations, and have provided mechanisms for users to fine-tune the ranking system to better suit their needs.

Usage Metrics: We now make usage metrics for ontologies available in the UI for ontology authors, and others to view. These metrics include the number of views of all ontologies in the previous month, and a graph of the number of views for a given ontology in the previous 18 months. We make these and other ontology access metrics available via the programmatic API.

CSV and RDF Downloads: We now allow users to download ontologies in both CSV and RDF formats. These features make it easier for users to manipulate ontologies in a spreadsheet (CSV) and/or to load them into a triple-store (RDF).

API Improvements: We have completely revised the API to be consistent across all of the functionality in BioPortal. We also now provide a number of API parameters that allow a user to specify exactly which fields to return. This feature can greatly speed up processing since the system does not have to look up and transfer unnecessary fields back to the client, only to then have the client discard them.

FUTURE PLANS

In the coming year we have plans for additional enhancements to BioPortal that will be outlined in the presentation.

ACKNOWLEDGEMENTS

The NCBO is one of the National Centers for Biomedical Computing supported by the NHGRI, the NHLBI, and the NIH Common Fund under grant U54-HG004028

REFERENCES

Musen, M. A., Noy, N. F., Shah, N. H., Whetzel, P. L., Chute, C. G., Story, M. A., Smith, B., and the NCBO team. The National Center for Biomedical Ontology. Journal of the American Medical Informatics Association. 2012 Mar-Apr; 19(2):190-5.

Whetzel, P. L., Noy, N. F., Shah, N. H., Alexander, P. R., Nyulas, C., Tudorache, T., and Musen, M. A. 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. 2011 Jul; 39 (Web Server issue):W541-5.

Salvadores, M., Horridge, M., Alexander, P. R., Fergerson, R. W., Musen, M. A., and Noy, N. F. Using SPARQL to Query BioPortal Ontologies and Metadata. In Proceedings of the International Semantic Web Conference (ISWC), 2012.

OWL-based form generation and structured data acquisition

Rafael S. Gonçalves*, Csongor I. Nyulas, Samson W. Tu, and Mark A. Musen
Stanford Center for Biomedical Informatics Research
Stanford University, Stanford, California, USA

ABSTRACT

We present a tool that is capable of generating Web forms from (question and answer) descriptions encoded in an OWL ontology. Unlike a regular form, the input fields of the generated form are associated with ontology concepts, and so the form is a means to acquire data to populate the ontology. The structure of this data is given by the modeling of questions and answers in the ontology, which makes the system flexible to different needs and goals. The tool is open-source, and freely distributed as a Web application.

1 SYSTEM DESCRIPTION

The Web Ontology Language (OWL) [4], being based on description logics (DL) [3], is not as amenable for structured data acquisition as a frame-based language; Protégé-Frames used definitions of classes in an ontology to generate knowledge-acquisition forms, which could be used to acquire instances of the classes [1]. This is not as straightforward with OWL, since class definitions are collections of axioms.

We describe a system that we implemented to: (a) generate Web forms from logical descriptions of questions and answers in an OWL ontology, and (b) acquire data from generated forms that is structured according to concepts in the ontology. We implemented our form generation and data acquisition tool mostly in Java, using the OWL API v4.0.1 [2].[1] The automatically-generated front-end of the form involves HTML, CSS and JavaScript. The source code of the tool is publicly available on GitHub.[2]

The inputs required from users in order to use this tool are: firstly, an OWL representation of the form structures (questions, sections, etc), and descriptions of the meaning of those structures (that is, whether the answer should be a string, integer, an OWL individual, etc.). We provide with our system a so-called *datamodel* ontology that users should extend in order to model their form(s), that is, user-defined questions should be inferred to be instances of *datamodel:Question*. Secondly, the view specification that is given by an XML file specifying user-interface aspects; for example, the organization of questions into sections, the order of questions, and more advanced options discussed further on. So, in order to use our software, a user will have to model questions and their descriptions in OWL, and then specify the layout and behavior of the resulting form in XML.

The tool takes as input the mentioned user-defined XML configuration (which should contain a pointer to the ontology specifying the content of the form, as well as pointers to imported ontologies), generates a Web form, and then parses and outputs form answers in CSV, RDF and OWL formats. The entire process is further described below.

(1) Form generation – Steps to produce a form:
  (a) Process XML configuration, gathering form layout information, IRIs and bindings to ontology entities
  (b) Extract from the input ontology all relevant information pertaining to each form element:
    (b.1) Text to be displayed (e.g., section header, question text)
    (b.2) Options and their text, where applicable
    (b.3) The focus of each question
  (c) Generate the appropriate HTML and JavaScript code
(2) Form input handling – Once the form is filled in and submitted:
  (a) Process answer data and create appropriate individuals
  (b) Produce a partonomy of the individuals created in (2.a) that mirrors the layout structure given in the configuration
  (c) Return the (structured) answers to the user in a chosen format

A key design choice of our system was to divide the specifications of user-interface aspects of the form (given by the XML file) and the content of the form (given by the OWL ontology). The user-defined XML configuration (1.a) specifies: input and output information of the tool, bindings to ontology entities, and layout of form elements. A document type definition (DTD) defines the building blocks of such configuration files, imposing necessary constraints to ensure the configuration file can be suitably interpreted. The key XML elements are:

input: contains an *ontology* child element, and optionally a child element named *imports*
- ***ontology***: absolute path or URL to the form specification ontology (e.g., *DBQ ontology*)
- ***imports***: contains *ontology* child elements, which have an attribute *iri*, giving the IRI of the imported ontology

output: contains the following child elements
- ***file***: defines, via a *title* attribute, the title of the form. Optionally, a path can be specified within the *file* element where the HTML form file should be serialized
- ***cssStyle***: the CSS style class to be used in the output HTML

bindings: defines mappings to ontology entities, such as what data property is used to state the text of a question, or section headings

form: defines the layout and behaviors of the form

More detailed implementation and configuration details can be found in the GitHub project wiki.

2 FEATURE SUMMARY

We briefly present the features of our system below.

Question triggering: a question can encode a key-value pair where the key is "showSubquestionsForAnswer" (or

*To whom correspondence should be addressed: rafaelsg@stanford.edu

[1] http://owlapi.sourceforge.net

[2] http://github.com/protegeproject/facsimile

"hideSubquestionsForAnswer") and the value is an IRI, which informs the view that when the answer corresponding to that IRI is selected, the question's subquestions should appear (or disappear, respectively).

Question types: the allowed question types in the generated form correspond to the HTML input-element types, with the addition of a pre-styled element: "checkbox-horizontal". By default checkbox inputs will be laid out vertically, hence the addition of the horizontal option.

Option ordering: answer options for a question can be given by an OWL enumeration, and our tool will order these options alphabetically by default. However, one may want to customize this order, perhaps to shift only one element or to re-order the whole set manually. This can be done in the definition of questions by inserting a key-value pair "orderOption" with the value being the desired order w.r.t. the default one. That is, if we want the (alphabetically-ordered) first element to appear last, we would have a value "*;1", which states: put the first element last, and everything else as it was.

Repeated question lists: each question list can be repeated a specified number of times, for example, in order to collect details of multiple family members.

Inline question lists: questions within "inline" question lists can be laid out horizontally rather than vertically (the default), by specifying the type of question list as "inline".

3 FUTURE PLANS

In the future we plan to make our software more versatile with the usage of XML Schema datatypes that are part of the OWL 2 specification datatype map. Another one of our goals is to design and implement a mechanism to facilitate the specification of forms, for instance, an interface to produce the required XML file.

ACKNOWLEDGMENTS

This work is supported in part by contract W81XWH-13-2-0010 from the U.S. Department of Defense, and grants GM086587 and GM103316 from the U.S. National Institutes of Health (NIH).

REFERENCES

[1] Eriksson, H., Puerta, A. R., and Musen, M. A. (1994). Generation of knowledge-acquisition tools from domain ontologies. *Int. J. of Human-Computer Studies*, **41**, 425–453.

[2] Horridge, M. and Bechhofer, S. (2009). The OWL API: A Java API for working with OWL 2 ontologies. In *Proc. of OWLED-09*.

[3] Horrocks, I., Kutz, O., and Sattler, U. (2006). The even more irresistible $\mathcal{SROIQ}$. In *Proc. of KR-06*.

[4] Motik, B., Patel-Schneider, P. F., and Parsia, B. (2009). OWL 2 Web Ontology Language: Structural specification and functional-style syntax. *W3C recommendation*.