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.

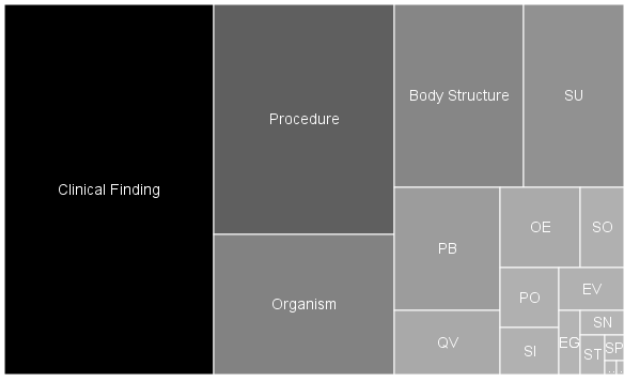


Fig. 1: SNOMED CT's shape represented with a treemap. Sub-hierarchies containing less than 10% of SNOMED CT concepts are shown in acronyms (see Table 1).

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.

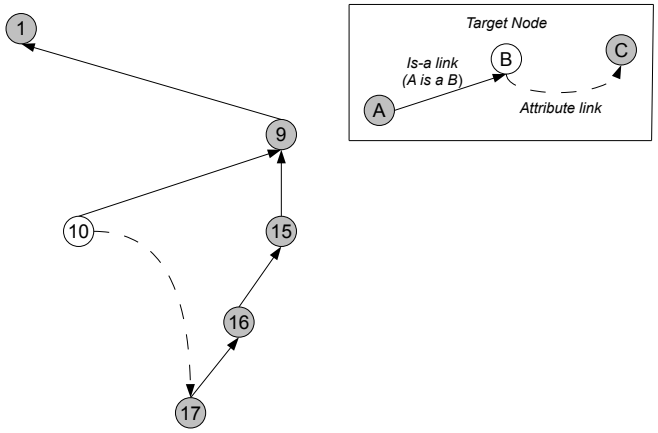


Fig. 2: Strategy followed to build our module M, starting from the seed concept (target node) 10. Figure 3 shows the original ontology from which it was extracted.

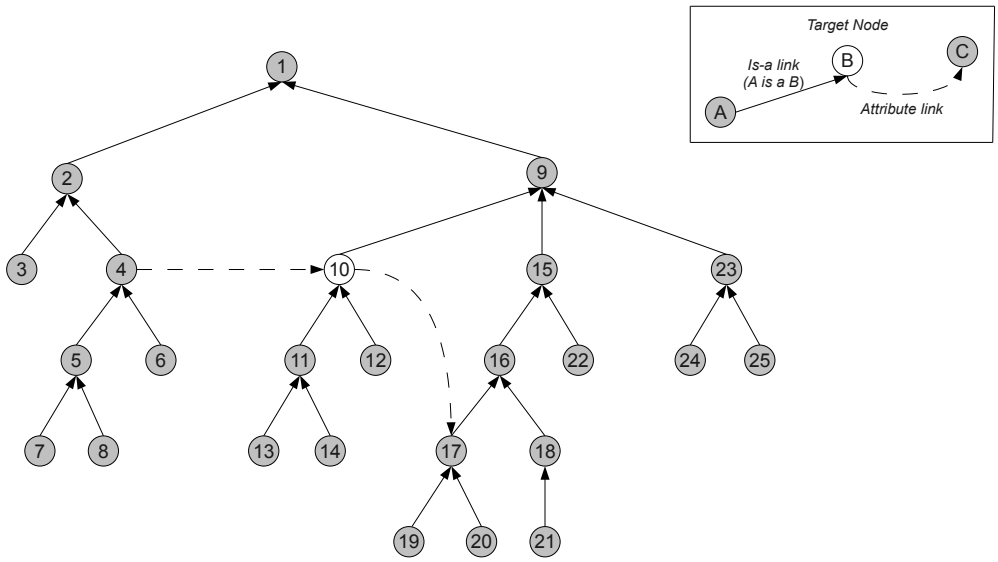


Fig. 3: Sample ontology, starting with a signature containing the seed node (target node) 10.

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 $RSS < 1$.

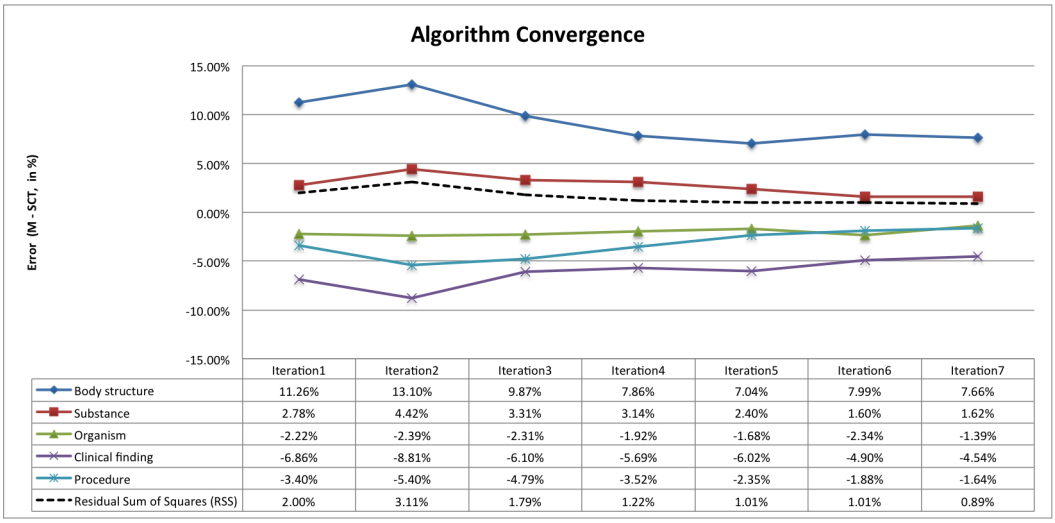
The algorithm, at each iteration i is the following:

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

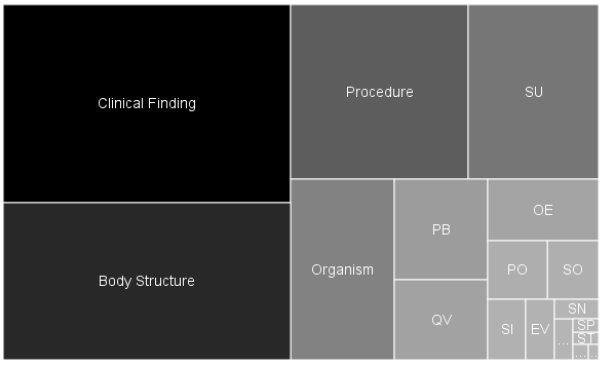
In our experiments, the algorithm converged after 7 iterations, extracting a module M with 10 834 classes. Figure 4 (Page 4) shows the error after each iteration for sub-hierarchies with more than 1% error, as well as the residual sum of squares.

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.

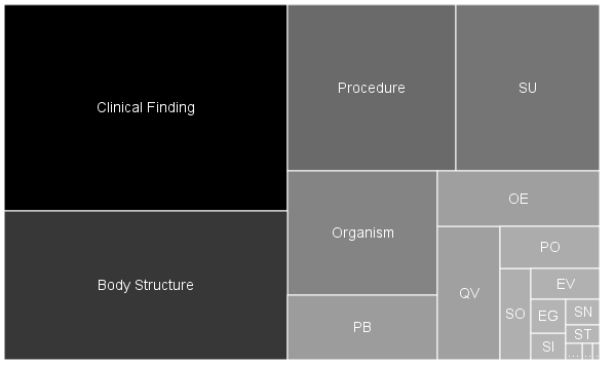


	Iteration1	Iteration2	Iteration3	Iteration4	Iteration5	Iteration6	Iteration7
Body structure	11.26%	13.10%	9.87%	7.86%	7.04%	7.99%	7.66%
Substance	2.78%	4.42%	3.31%	3.14%	2.40%	1.60%	1.62%
Organism	-2.22%	-2.39%	-2.31%	-1.92%	-1.68%	-2.34%	-1.39%
Clinical finding	-6.86%	-8.81%	-6.10%	-5.69%	-6.02%	-4.90%	-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%

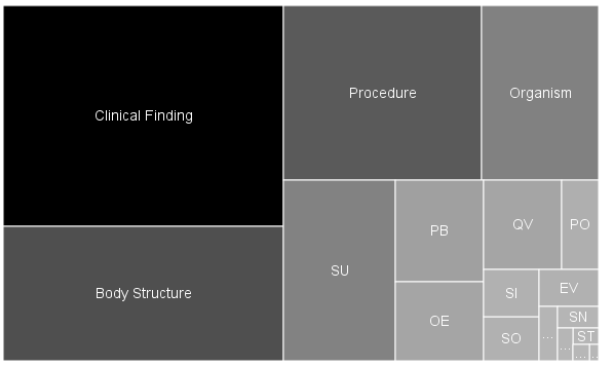
Fig. 4: Execution of the algorithm, showing convergence in iteration 7.



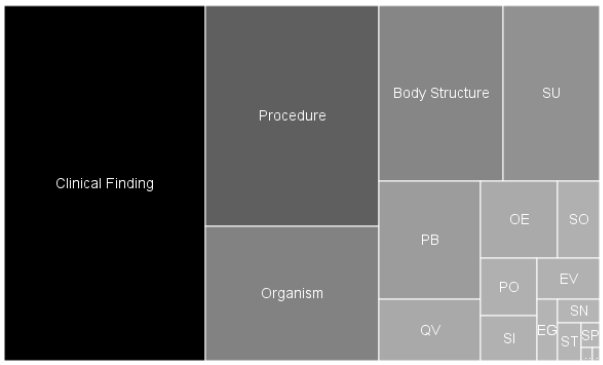
(a) Module Shape - Iteration 1



(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.