A Survey of Syntactic Modelling Structures in Biomedical Ontologies

Christian Kindermann and Martin G. Skjæveland

Department of Informatics, University of Oslo
{chrikin,martige}@ifi.uio.no

Abstract. Despite the large-scale uptake of semantic technologies in the biomedical domain, little is known about common modelling practices in published ontologies. OWL ontologies are often published only in the crude form of sets of axioms leaving the underlying design opaque. However, a principled and systematic ontology development life cycle is likely to be reflected in regularities of the ontology's emergent syntactic structure. To develop an understanding of this emergent structure, we propose to reverse-engineer ontologies taking a syntax-directed approach for identifying and analysing regularities for axioms and sets of axioms. We survey BioPortal in terms of syntactic modelling trends and common practices for OWL axioms and class frames. Our findings suggest that biomedical ontologies only share simple syntactic structures in which OWL constructors are not deeply nested or combined in a complex manner. While such simple structures often account for large proportions of axioms in a given ontology, many ontologies also contain non-trivial amounts of more complex syntactic structures that are not common across ontologies.

1 Introduction

The uptake of OWL in the biomedical domain has lead to the development of a large number of ontologies as well as tools providing support for ontology construction and maintenance. While some ontologies are documented to follow pattern-based design principles, e.g., [19,21], little is known about what kind of design choices, principles, and patterns are widely-used, how they impact ontology engineering in practice. Comparing ontologies in terms of their design rationales is often challenging because different ontology are developed and maintained using a wide range of methodologies, techniques, and tools. Moreover, ontologies are often published as a single file with scarce to no documentation. Yet, a principled and systematic ontology design is likely to be reflected in regularities of the ontology's emergent syntactic structure.

So, to develop an understanding of common practices in ontology engineering, we propose to *reverse-engineer* ontologies in terms of syntactic regularities. Identified regularities may then be analysed and compared to distil common modelling structures both within and across ontologies. In this work, we focus on the syntactic structure of logical expressions in OWL ontologies. In particular, we analyse the way they are composed and combined. The contributions are as follows: (i) we adapt and simplify the formal framework for identifying syntactic regularities originally proposed in [9,10], (ii) we extend this framework by developing methods for analysing such regularities

w.r.t. their underlying syntactic structures, and (iii) we conduct an empirical study to characterise the syntactic structure of axioms and class frames in biomedical ontologies.

This paper is accompanied by a technical report [11] providing more detailed examples, an in-depth discussion about differences between this and prior work, and a more elaborate presentation of both the motivation and potential impact of our work.

2 Preliminaries

We assume the reader to be familiar with Description Logics (DL) [1] and the Web Ontology Language (OWL) [5]. We use DL notation for the sake of readability but interpret logical constructors as specified by OWL. Furthermore, we use both infix and prefix notation for presentational purposes, e.g., $\mathit{SubClassOf}(\mathsf{A}, \mathsf{B})$ may be written as $\mathsf{A} \sqsubseteq \mathsf{B}$ or $\sqsubseteq(\mathsf{A}, \mathsf{B})$. We disregard OWL annotations, i.e., axioms with and without annotations are indistinguishable.

A directed labelled *graph* g is an ordered pair (N, E, L) where N is a set of nodes, L is a set of labels, and $E \subseteq N \times L \times N$ is a set of edges. A graph $s = (N', E', L')$ is a *subgraph* of g, written $s \lesssim g$, if $N' \subseteq N$ an $E' \subseteq E$. A *graph isomorphism* between two graphs $g_1 = (N_1, E_1, L_1)$ and $g_2 = (N_2, E_2, L_2)$ is a bijection $f : N_1 \cup L_1 \to N_2 \cup L_2$ s.t. $(n, l, n') \in E_1$ iff $(f(n), f(l), f(n')) \in E_2$. Two graphs are isomorphic if there exists an isomorphism between them. A *contraction* of an edge $e = (n_1, l, n_2) \in E$ with $n_1 \neq n_2$ is an operation that first removes e from E and replaces both n_1 and n_2 with a single node n' and then makes any node (originally) adjacent to either n_1 or n_2 adjacent to n'. A minor of a graph is a graph obtained by (iteratively) contracting edges, removing edges, or removing nodes without adjacent nodes.

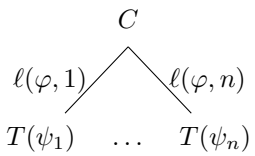
3 Framework for Syntax-Directed Analysis of OWL Ontologies

3.1 Syntactic Regularities

We analyse structures in OWL ontologies using a syntax-directed approach based on their abstract representation according to the structural specification for OWL 2 [16]. This abstract representation can be captured by abstract syntax trees (AST).

Definition 1 (OWL Abstract Syntax Tree). *Let φ be an OWL expression. Then, the abstract syntax tree for $T(\varphi)$ is defined as follows:*

- *if φ is atomic, then $T(\varphi)$ is a node labelled with φ,*
- *if $\varphi = C(\psi_1, \ldots, \psi_n)$, where C is an OWL constructor and $\psi_1, \ldots, \psi_n$ are OWL expressions, then $T(\varphi) =$*



where ℓ is a labelling function for branches s.t. $\ell(\varphi, i)$ specifies how a subexpression ψ_i at position i is used in relation to C.

The labelling function ℓ is used to treat abstract syntax trees for OWL expressions uniformly as *unordered* trees even in cases where the order of arguments for OWL constructors matters. Consider for example the AST of $\mathit{SubClassOf}(\mathsf{A}, \mathsf{B})$. Here the branches to A and B would be labelled with "Subclass" and "Superclass" respectively. In the following, we will not distinguish between OWL axioms and their ASTs, i.e., an axiom will be referred to simply as a tree (meaning its AST) and vice versa. Similarly, an ontology can be understood as a set of trees.

Given the notion of OWL abstract syntax trees, we can formulate syntax-directed *transformations* for OWL abstract syntax trees that highlight specific syntactic properties of OWL expressions. In particular, we can highlight *shared* syntactic properties between OWL axioms to identify recurring expressions. Consider the axioms $\alpha_1 = \mathsf{A}_1 \sqsubseteq \exists\, \mathsf{P}.\mathsf{A}_2$ and $\alpha_2 = \mathsf{B}_1 \sqsubseteq \exists\, \mathsf{Q}.\mathsf{B}_2$. While both axioms differ in terms of named classes and properties, they coincide otherwise. This structural similarity can be highlighted via a syntax-directed transformation that *abstracts* over syntactic properties in which two axioms differ. For example, with a transformation G that replaces atomic entities with a placeholder symbol, say $*$, we have $G(\alpha_1) = G(\alpha_2) = * \sqsubseteq \exists * . *$. Put differently, α_1 and α_2 exhibit the same syntactic structure that is preserved under the *abstraction* G. An abstraction is intuitively understood as an operation that *hides* some level of detail. This intuition can be captured for transformations of ASTs by restricting them to the removal of branches and nodes.

Definition 2 (Language Abstraction). *An* abstraction *for a tree language $\mathcal{L}$ into a tree language $\mathcal{L}'$ is defined by a function $A\colon \mathcal{L} \to \mathcal{L}'$ such that*

1. *there exist $t, t' \in \mathcal{L}$ s.t. $t \neq t'$ with $A(t) = A(t')$,*
2. *for $t \in \mathcal{L}$ there exists a graph minor t_m that is isomorphic to $A(t)$.*

The second condition formalises the idea of only allowing the removal of a tree's branches and nodes whereas the first condition requires that an abstraction hides some kind of information so that two syntax trees become *indistinguishable*. Coming back to the earlier observation that $G(\alpha_1) = G(\alpha_2)$, we note that axiom equality under a given abstraction gives rise to an equivalence relation w.r.t. the syntactic structure of axioms in an ontology. We refer to corresponding equivalence classes as *syntactic regularities*.

Definition 3 (Syntactic Regularity for Axioms). *A* syntactic regularity for axioms *in an ontology $\mathcal{O}$ is an equivalence class $[\alpha]_A = \{\alpha_i \in \mathcal{O} \mid A(\alpha_i) = A(\alpha)\}$, where A is a language abstraction.*

While axioms are the primary building blocks in OWL ontologies, an entity is often not represented by single axiom but by a *set* of axioms. So, in addition to regularities for axioms, we are also interested in regularities for sets of axioms. We defer the discussion of how to group related axioms into sets until Section 3.2. Here, we only note that the notion of syntactic regularities for axioms can be lifted to sets of axioms in a straightforward way. By abuse of notation, we write $A(S)$ to denote a language abstraction on *forests* of syntax trees S rather than syntax trees only.

Definition 4 (Syntactic Regularity for Sets of Axioms). *Let $\mathcal{S} = \{S_1, \ldots, S_n\}$ be a family of sets of axioms in an ontology $\mathcal{O}$. A* syntactic regularity for sets of axioms *in $\mathcal{O}$ w.r.t. $\mathcal{S}$ is an equivalence class $[S]_A = \{S_i \in \mathcal{S} \mid A(S) = A(S_i)\}$ where A is a language abstraction.*

$\mathcal{O} = \{\ \alpha_1 = \mathsf{A} \sqsubseteq \sqcap(\mathsf{A}_1, \mathsf{A}_2),$
$\alpha_2 = \mathsf{B} \sqsubseteq \sqcap(\mathsf{B}_1, \mathsf{B}_2),$
$\alpha_3 = \mathsf{C} \sqsubseteq \sqcap(\mathsf{C}_1, \mathsf{C}_2, \mathsf{C}_3)\ \}$

$[\alpha_1]_G = \{\alpha_1, \alpha_2\}$
$[\alpha_3]_G = \{\alpha_3\}$
$[\alpha_1]_I = \{\alpha_1, \alpha_2, \alpha_3\}$

(a)

(b) (c)

Fig. 1: Example of the language abstractions G and I applied to a sample ontology, and their associated modelling structures: (a) shows the sample ontology (of three axioms) and its syntactic regularities under G and I, (b) displays the two modelling structures for $\mathcal{O}$ under G, while (c) shows the single modelling structure for $\mathcal{O}$ under I. Branch labels are not shown.

3.2 Modelling Structures

A syntactic regularity w.r.t. a language abstraction is uniquely determined by an abstract syntactic structure, namely the abstract syntax tree or forest that each of its elements are mapped to under the used language abstraction. We will refer to these abstract structures as *modelling structures*.

Definition 5 (Modelling Structure). *Let $\mathcal{O}$ be an OWL ontology, $\alpha \in \mathcal{O}$, and $S \subseteq \mathcal{O}$, and A a language abstraction. Then $A(\alpha)$ and $A(S)$ are* modelling structures *for α and S under A respectively.*

So, a language abstraction gives rise to syntactic regularities in an ontology and each syntactic regularity is associated with a modellling structure. In the following, we provide concrete examples for these notions. We already mentioned the language abstraction G that highlights structural similarities between axioms by abstracting over atomic entities. We will refer to this abstraction as the ground generalisation.

Definition 6 (Ground Generalisation). *Let t be an OWL abstract syntax tree. The* Ground Generalisation *$G(t)$ of t is a language abstraction defined by a function G that replaces the label of each leaf node in t with the label $*$.*

The example ontology in Figure 1(a) has two syntactic regularities w.r.t. G, namely $[\alpha_1]_G = \{\alpha_1, \alpha_2\}$ and $[\alpha_3]_G = \{\alpha_3\}$, which each give rise to a modelling structure under G, shown in Figure 1(b): $G(\alpha_1) = G(\alpha_2) = * \sqsubseteq \sqcap(*, *)$ and $G(\alpha_3) = * \sqsubseteq \sqcap(*, *, *)$. Note that we use prefix notation for the n-ary constructor $\sqcap$ to avoid notational ambiguity. However, all three axioms in the example can be characterised in terms of the nesting of OWL constructors, i.e., all three are subsumption axioms with a conjunction on the right-hand side. The nesting of constructors in OWL axioms can be distilled with a transformation that removes all leaf nodes (and corresponding branches) from the axiom's associated abstract syntax tree. We will refer to the nesting structure of OWL constructors as an axiom's internal tree structure.

Definition 7 (Internal Tree Structure). *Let t be an OWL abstract syntax tree. The* internal tree structure *$I(t)$ of t is a language abstraction defined by a function I that removes all leaf nodes and corresponding branches from t.*

The example ontology in Figure 1(a) has only one syntactic regularity w.r.t. I, shown in Figure 1(c), since $I(\alpha_1) = I(\alpha_2) = I(\alpha_3)$. Intuitively, the abstraction I abstracts over more syntactic properties compared to G which leads to fewer but larger syntactic regularities (where the size of a regularity is the number of its elements, i.e., axioms).

As already mentioned in Section 3.1, conceptual models for domain-specific entities are, more often than not, represented with a *set of axioms* rather than with a single axiom. The notion of a *class frame* is widely used for grouping conceptually related axioms in OWL ontologies [7,18].

Definition 8 (Class Frame). *A* class frame $CF(\mathsf{C}, \mathcal{O})$ *for a class expression* C *in an ontology* $\mathcal{O}$ *is defined as the set:* $CF(\mathsf{C}, \mathcal{O}) = \{\alpha \in \mathcal{O} \mid \alpha = SubClassOf(\mathsf{C}, \mathsf{C}'),$ *or* $\alpha = EquivalentClasses(\mathsf{C}, \mathsf{C}_1, \ldots, \mathsf{C}_n)\},$ *or* $\alpha = DisjointClasses(\mathsf{C}, \mathsf{C}_1, \ldots, \mathsf{C}_n)\},$ *or* $\alpha = DisjointUnion(\mathsf{C}, \mathsf{C}_1, \ldots, \mathsf{C}_n)\}$.

The abstractions I and G for abstract syntax trees of axioms can be lifted to forests of abstract syntax trees in a straightforward manner.

Definition 9 (Multiset Lifting of Language Abstractions). *Let* F *be a forest of OWL abstract syntax trees and* A *a language abstraction for OWL abstract syntax trees. Then the image* $A(F)$ *of* F *under* A *is defined as the multiset* $A(F) = \{A(t) \mid t \in F\}$.

We define $A(F)$ as a multiset to account for repetitions of axioms with the same modelling structure. Consider the set $F = \{SubClassOf(\mathsf{C}, \mathsf{B}), SubClassOf(\mathsf{C}, \mathsf{D})\}$. Using a set for the lifiting of G would yield $\{SubClassOf(*, *)\}$ instead of the desired multiset. We write α^x to denote the x-fold repetition of modelling structure α. So, $\{SubClassOf(*, *)^2\}$ denotes the multiset $\{SubClassOf(*, *), SubClassOf(*, *)\}$.

3.3 Relations between Modelling Structures

The intention of G with regards to syntactic regularities is to group OWL axioms or sets of axioms based on the way OWL constructors are combined and nested. In particular, any difference between axioms in terms of used OWL constructors will be captured by different syntactic regularities. Consider the axioms $\alpha_1 = \mathsf{A} \sqsubseteq \exists\ \mathsf{R}.\mathsf{B}$ and $\alpha_2 = \mathsf{A} \sqsubseteq \exists\ \mathsf{R}.(\exists\ \mathsf{R}.\mathsf{B})$. Clearly, $G(\alpha_1) \neq G(\alpha_2)$. Note, however, that the nesting of OWL constructors in α_1, i.e., its internal tree structure $I(\alpha_1)$, occurs as a *substructure* in α_2. We can formalise this substructure relationship via subgraphs in modelling structures.

Definition 10 (Structure Containment). *Let* t *and* t' *be two OWL abstract syntax trees. Then,* t structurally contains t'*, written* $t \lesssim_I^G t'$*, if*

1. $I(t) \lesssim I(t')$ *and* $I(t) \neq I(t')$*, or*
2. $G(t) \lesssim G(t')$ *and* $I(t) = I(t')$.

The two cases in the definition for structure containment are owed to n-ary constructors. In the case of two OWL expressions e and e' that only involve constructors with a fixed arity we have that $I(e) = I(e')$ implies $G(e) = G(e)$. However, this is not the case for expressions involving n-ary constructors. Consider for example the axioms $\alpha_1 = \mathsf{A} \sqsubseteq \sqcap(\mathsf{C}_1, \mathsf{C}_2)$ and $\alpha_2 = \mathsf{A} \sqsubseteq \sqcap(\mathsf{C}_1, \mathsf{C}_2, \mathsf{C}_3)$. Here, we have $I(\alpha_1) = I(\alpha_2)$

but $G(\alpha_1) \neq G(\alpha_2)$. So, defining the substructure containment between OWL abstract syntax trees only in terms of their internal tree structures would ignore structural information about n-ary constructors. The second case in Definition 10 rectifies this so that α_2 structurally contains α_1. The structure containment relation defines a partial order on OWL abstract syntax trees and thus induces a partial order on syntactic regularities for axioms.

Lemma 1 (Partial Order on Ground Generalisations). *Let $[t_1]_G, \ldots, [t_n]_G$ be syntactic regularities for axioms w.r.t. G in an ontology $\mathcal{O}$. Then the relation $\lesssim_I^G$ induces a partial order on $[t_1]_G, \ldots, [t_n]_G$.*

Similarly, we can induce a partial order on syntactic regularities for class frames w.r.t. G by defining a containment relation based on a notion of subsets for multisets. That is, for each number of axioms with the same ground generalisation in one class frame there needs to exist at least as many axioms with an identical ground generalisation in the other class frame.

Definition 11 (Class Frame Containment). *Let C and C' be class frames in an ontology $\mathcal{O}$. If there exists an injective mapping $m\colon C \to C'$ s.t. $t \in C$ implies that $G(t) = G(m(t))$, then C'* contains *C, written $C \lesssim_G C'$.*

Lemma 2 (Partial Order on Class Frames). *Let $[C_1], \ldots, [C_n]$ be syntactic regularities for class frames in an ontology $\mathcal{O}$. Then the relation $\lesssim_G$ for class frames induces a partial order on $[C_1], \ldots, [C_n]$.*

4 Methods

Research Questions. To develop a first understanding of syntactic structures in published ontologies, we focus on properties related to OWL constructors for class expressions. In particular, we investigate to what extent such constructors are nested and combined to give rise to more complex structures. Furthermore, we aim to identify and characterise common structures within and across ontologies. Lastly, we investigate to what extent distinct syntactic structures are related by shared substructures.

Experimental Design. Since we are interested in the way OWL constructors are used in OWL ontologies, we will investigate syntactic regularities w.r.t. the language abstraction G proposed in Section 3.2. So, we will refer to syntactic regularities based on G (for axioms and class frames) simply as regularities (for axioms and class frame respectively) unless stated otherwise. Likewise, we will not explicitly specify that modelling structures for regularities are based on G unless the context is ambiguous. Our investigation consists of five experiments. In the following, we give a brief description for each of these experiments and describe the construction of the experimental corpus of ontologies using BioPortal. We refer the interested reader to the technical report [11] for a discussion of using BioPortal for the purposes of this study.

1. Number of Syntactic Regularities. We determine to what extent ontologies give rise to different regularities, i.e., contain different syntactic structures.

2. Size of Syntactic Regularities. We give an account of the size of syntactic regularities. Since a regularity is a set, its size is defined by the number of its elements.

3. Characteristics of Common Modelling Structures. We determine what kind of modelling structures are common within and across ontologies. For this purpose, we inspect the three largest syntactic regularities in each ontology and qualify their associated modelling structures in terms of the nesting and combination of OWL constructors. Furthermore, we compare the modelling structures associated with large regularities across ontologies to identify structures of a general nature.

4. Size and Depth of Modelling Structures. We determine to what extent OWL constructors are nested and combined in modelling structures. For this purpose, we report on the maximal size and depth of modelling structures in ontologies. Since a modelling structure for axioms is a tree, its depth is defined as its tree depth, i.e., the longest path from its root to a child. In the case of modelling structures for class frames, their depth is defined as the maximal depth of its axioms.

5. Interrelations between Syntactic Regularities. We determine to what extent syntactic regularities in ontologies are structurally related. So, we analyse the partially ordered sets of syntactic regularities w.r.t. the notions of structural containment (cf. Section 3.3). In particular, we construct the Hasse diagrams associated with said posets for each ontology and report on their longest paths, i.e, their depth, as well as their maximal branching factors.

Ontology Corpus. We work with a recent (February 2022) snapshot of BioPortal created in the same way as described in [14]. The data set of ontologies encompasses a total of 736 ontologies. We use the OWL API[1] (v.5.1.15) to orchestrate all experiments. Therefore, we restrict the experimental corpus to ontologies that can be loaded with the OWL API. We load ontologies without their imports closure to avoid double counting syntactic structures that are imported by different ontologies. Furthermore, we exclude ontologies that do not contain class expression axioms because our experiments are restricted to class expression axioms. Lastly, we exclude ontologies for which we could not compute all syntactic regularities and their interrelations within one hour. This procedure results in an experimental corpus of 657 ontologies.

In our experiments, we distinguish between three kinds of ontologies. First, ontologies that consist of atomic axioms only, i.e., $SubClassOf$ and $EquivalentClasses$ axioms that have only named classes as arguments. Second, ontologies expressible in $\mathcal{EL}^{++}$. And third, ontologies not expressible in $\mathcal{EL}^{++}$. We refer to these three kinds of ontologies as atomic, $\mathcal{EL}^{++}$, and rich ontologies respectively. Figure 2 shows the size of an ontology's TBox as well as the size of its subset of class expression axioms. We order ontologies within a category by size and assign each ontology an index in ascending order starting with atomic ontologies as shown in Figure 2. The corpus contains 94 atomic ontologies, 90 $\mathcal{EL}^{++}$ ontologies, and 473 rich ontologies.

5 Results

We present results for the five experiments as specified in Section 4 in separate subsections. We remind the reader that our experimental design distinguishes between three categories of ontologies (atomic, $\mathcal{EL}^{++}$, and rich) and that we have two experimental

[1] `http://owlcs.github.io/owlapi/`

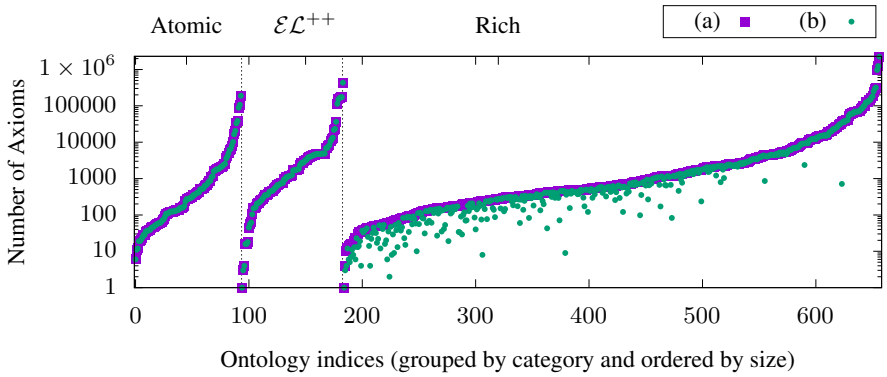


Fig. 2: Number of TBox axioms (a) and class expression axioms (b).

conditions for all three categories, namely, (a) regularities for axioms and (b) regularities for class frames.

5.1 Experiment 1: Number of Syntactic Regularities

The number of different syntactic regularities for (a) axioms and (b) class frames are shown in Figure 3 for all three categories of ontologies.

The data reveals that atomic and $\mathcal{EL}^{++}$ ontologies give rise to mostly only one or two regularities for axioms whereas rich ontologies give rise to varying numbers of regularities for axioms. While the largest number of regularities can be found in large rich ontologies, it is not the case that all large ontologies give rise to many regularities.

Even though atomic and $\mathcal{EL}^{++}$ ontologies exhibit only a few regularities for axioms and thus contain mostly axioms of the same syntactic structure, these axioms are combined in many ontologies to give rise to a comparatively larger number of regularities for class frames. For example, the $\mathcal{EL}^{++}$ ontology RH-MESH at index 183 has only two regularities for axioms but 65 regularities for class frames. Similarly, most rich ontologies, especially larger ones beyond index 351 (with about 350 axioms), often give rise to considerably more regularities for class frames compared to regularities for axioms. For example, the rich ontology FMA at index 652 gives rise to 99 regularities for axioms and 3487 regularities for class frames.

5.2 Experiment 2: Size of Syntactic Regularities

The results of Experiment 1 show that many rich ontologies give rise to a fair number of regularities for axioms. In [10], the same result was found for an older snapshot of BioPortal and it was reported that only a few of these regularities for axioms are large. In particular, in the case of regularities for axioms, it was determined that 90% of axioms in many ontologies can be covered by one to three regularities in all three ontology categories. However, the same could not be reported for regularities of class frames;

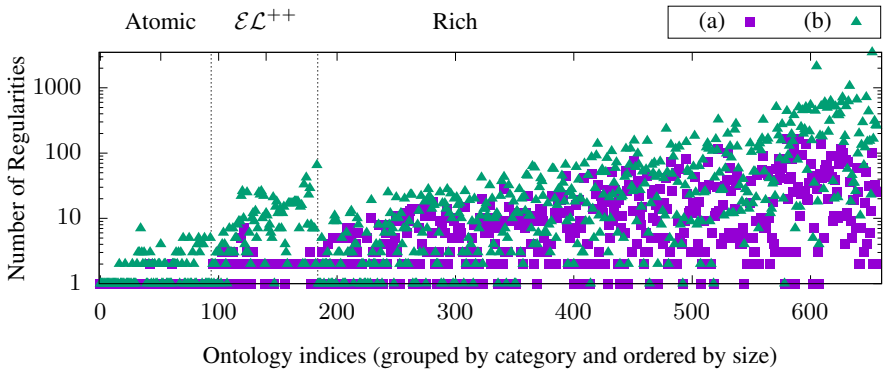


Fig. 3: Number of regularities (with respect to G) for (a) axioms and (b) class frames in atomic, $\mathcal{EL}^{++}$, and rich ontologies.

Table 1: Number of ontologies giving rise to a minimal number of regularities (both for axioms and class frames) with a minimal size of 10, 100, and 1000.

Min. Regularities	**Min. Size**	**Number of Ontologies**					
		Regularities for Axioms			Regularities for Class Frames		
		Atomic	$\mathcal{EL}^{++}$	Rich	Atomic	$\mathcal{EL}^{++}$	Rich
5	10	-	-	127	1	37	189
	100	-	-	35	1	6	64
	1000	-	-	8	-	1	21
10	10	-	-	45	-	9	108
	100	-	-	7	-	1	34
	1000	-	-	-	-	-	5

especially for larger rich ontologies. In the case of class frames, it was reported that often more than ten regularities are required to account for 90% of axioms in a given ontology.

While this finding gives some indication for the size of the three largest regularities in ontologies, it is important to keep in mind that many ontologies in our experimental corpus contain several thousands of axioms and that small relative proportions of an ontology can still correspond to many axioms. So, to give an account of the size of regularities in terms of absolute numbers, we report on the number of ontologies that contain at least five or ten regularities with a minimal size of (i) ten, (ii) a hundred or (iii) a thousand elements in Table 1.

It transpires that mostly rich ontologies give rise to multiple regularities of non-trivial sizes within a given ontology. In the case of regularities for axioms, for example, there are 35 rich ontologies with at least 5 regularities that have at least 100 elements. In the case of regularities for class frames, there are even 34 rich ontologies with at least 10 regularities that have at least 100 elements. This confirms to some extent the hypothesis that there exist ontologies with more than three regularities of non-trivial size. However, increasing either the number of minimal regularities, e.g., to ten, or the number of

minimal elements, e.g., to 1000, reveals that there are only a few ontologies with many regularities of considerable size.

Lastly, we note that many rich ontologies *do not* give rise to at least 5 regularities with a minimal size of ten. This is interesting in the context of the total number of ontologies (cf. Section 5.1) that give rise to 5 or more regularities. In the case of regularities for axioms, there are 285 such rich ontologies which means that $285 - 127 = 158$ ontologies contain only a few large regularities despite giving rise to 5 or more. Similarly, in the case of regularities for class frames, there are $364 - 189 = 175$ such ontologies.

5.3 Experiment 3: Characteristics of Common Modelling Structures

We remind the reader that each syntactic regularity is associated with a unique modelling structure. So, we can identify common syntactic structures *within* an ontology by inspecting the modelling structures of the ontology's largest regularities. Furthermore, we can identify common syntactic structures *across* ontologies by comparing modelling structures associated with the largest regularities within ontologies.

The three largest regularities for axioms across atomic, $\mathcal{EL}^{++}$, and rich ontologies give rise to 2, 11, and 103 distinct modelling structures respectively. Table 2 lists those modelling structures[2] that occur across at least 20 different ontologies. The values in the last three columns of Table 2 reveal the actual number of ontologies in which a given modelling structure is associated with one of the three largest regularities, e.g., the modelling structure $EquivalentClasses(*,*)$ is associated with one of the three largest regularities in two atomic ontologies, two $\mathcal{EL}^{++}$ ontologies, and 24 rich ontologies.

Overall, it transpires that only a few modelling structures for axioms are common both within and across ontologies. Furthermore, these modelling structures are fairly simple in regards to the way OWL constructors are nested and combined. Nevertheless, it is important to keep in mind that rich ontologies exhibit a large variety of modelling structures that are associated with their respective largest regularities. It is also important to mention that many such structures are more complex compared to the ones shown in Table 2. For example, the second largest regularity in the ontology HOOM with 78738 elements is associated with the modelling structure

$$EquivalentClasses(*,\ ObjectIntersectionOf(ObjectSomeValuesFrom(*,*),\ ObjectSomeValuesFrom(*,*),\ ObjectSomeValuesFrom(*,*),\ ObjectSomeValuesFrom(*,*),\ DataHasValue(*,*))).$$

So, while common modelling structures for axioms *across* ontologies are mostly simple, common modelling structures *within* ontologies can also be rather complex.

The three largest regularities for class frames across atomic, $\mathcal{EL}^{++}$, and rich ontologies give rise to 6, 28, and 209 distinct modelling structures respectively. Table 3 lists those modelling structures for class frames that occur across at least 20 different ontologies in the same manner as Table 2 lists modelling structures for axioms. The results are similar to the case for regularities for axioms in the sense that common modelling structures for class frames across ontologies are mostly simple, i.e., the class

[2] The prefix "Object" in some OWL expressions is abbreviated with the capital letter "O" for presentational purposes.

Table 2: Common modelling structures across ontologies. A modelling structure is considered common in a given ontology if it associated with one of its three largest regularities. Ordered by total number of ontologies.

Row	Modelling Structure	Atomic	$\mathcal{EL}^{++}$	Rich
1	$SubClassOf(*,*)$	94	88	466
2	$SubClassOf(*, OSomeValuesFrom(*,*))$	-	68	270
3	$DisjointClasses(*,*)$	-	-	103
4	$EquivalentClasses(*, OIntersectionOf(*, OSomeValuesFrom(*,*)))$	-	1	70
5	$SubClassOf(*, OAllValuesFrom(*,*))$	-	-	44
6	$EquivalentClasses(*,*)$	2	2	24
7	$SubClassOf(*, OExactCardinality(*,*,*))$	-	-	20

Table 3: Number of ontologies in which its the three largest regularities for class frames is associated with a given modelling structure. Ordered by total number of ontologies.

Row	Modelling Structure	Atomic	$\mathcal{EL}^{++}$	Rich
1	$\{SubClassOf(*,*)^1\}$	94	67	431
2	$\{SubClassOf(*,*)^2\}$	37	32	106
3	$\{SubClassOf(*,*)^3\}$	16	15	20
4	$\{SubClassOf(*, OSomeValuesFrom(*,*))^1\}$	-	22	12
5	$\{EquivalentClasses(*, OIntersectionOf(*, OSomeValuesFrom(*,*)))^1\}$	-	1	62
6	$\{DisjointClasses(*,*)^1\}$	-	-	22
7	$\{SubClassOf(*,*)^1, SubClassOf(*, OSomeValuesFrom(*,*))^1\}$	-	35	157
8	$\{SubClassOf(*,*)^1, SubClassOf(*, OSomeValuesFrom(*,*))^2\}$	-	9	50
9	$\{SubClassOf(*,*)^1, SubClassOf(*, OSomeValuesFrom(*,*))^3\}$	-	15	8
10	$\{SubClassOf(*,*)^1, DisjointClasses(*,*)^1\}$	-	-	58
11	$\{SubClassOf(*,*)^1, EquivalentClasses(*,*)^1\}$	2	-	19

frames consist of only a few axioms and the axioms are not deeply nested. Likewise, there are also many ontologies in which the largest three regularities for class frames are associated with more complex modelling structures involving more axioms or more deeply nested OWL constructors (see regularities in CLO for example). However, such more complex modelling structures are only common within ontologies and not across.

5.4 Experiment 4: Size and Depth of Modelling Structures

In this section, we shed some light on the most complex modelling structures in ontologies. We start with the size of modelling structures, i.e., their number of nodes. Figure 4 shows the size of the largest modelling structures in ontologies for both (a) axioms and (b) class frames. We will first highlight some details about the size of modelling structures for axioms before we compare them to modelling structures for class frames.

The maximal size of modelling structures for axioms in atomic ontologies is three because they only contain the modelling structures $* \sqsubseteq *$ and $* \equiv *$. Similarly, the size of modelling structures in most $\mathcal{EL}^{++}$ontologies is three or five because they only contain the modelling structures $* \sqsubseteq *$ and $* \sqsubseteq \exists * . *$. There are only four ontologies containing modelling structures with a size larger than five. The largest one is found in the ontology CHIRO with size 11 and has the form $* \equiv * \sqcap (\exists * .(* \sqcap (\exists * .*)))$.

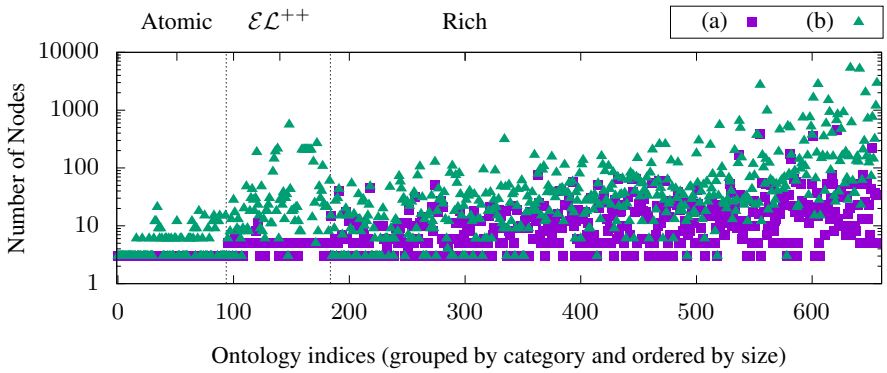


Fig. 4: Number of nodes in the largest modelling structures associated with regularities for (a) axioms and (b) class frames.

However, about half of rich ontologies (211 out of 473) contain modelling structures for axioms with a size larger than ten. Interestingly, the maximal size of modelling structures in ontologies appears be independent of the ontologies' overall size, i.e., modelling structures of different sizes occur in ontologies of different sizes.

The maximal size of modelling structures for class frames is often considerably larger compared to the maximal size of modelling structures for axioms, especially for $\mathcal{EL}^{++}$and rich ontologies that have more than about 350 axioms. This is to be expected if class frames consist of combinations of many axioms. In this regard, it transpires that class frames in many atomic ontologies and many rich ontologies of smaller size consist of only single axioms. On the right-hand side of Table 4, we summarise how many ontologies contain class frames up to a maximal number axioms. It appears that $\mathcal{EL}^{++}$and rich ontologies contain class frames with more than three axioms whereas many atomic ontologies only contain class frames with one or two axioms.

In addition to the size of modelling structures, we also investigate their depth. Note that the depth of a class frame is defined in terms of the maximal depth of its axioms. So, the maximal depth of modelling structures for both axioms and class frames is the same and we will not distinguish between the two in the following. On the left-hand side of Table 4, we summarise how many ontologies contain modelling structures up to a maximal depth. There are 167 rich ontologies that contain modelling structures with a depth of at least four. This shows that many rich ontologies not only contain fairly large modelling structures but that modelling structures also involve non-trivial nestings of OWL constructors.

5.5 Experiment 5: Interrelations between Syntactic Regularities

Table 5 shows the depth and maximal branching factor of Hasse diagrams corresponding to partially ordered sets for syntactic regularities for axioms and class frames w.r.t. $\lesssim_I^G$ and $\lesssim_G$ respectively. It transpires that more than half of the ontologies in our

experimental corpus (365 out of 657) give rise to Hasse diagrams with a depth of at least 4. Moreover, 110 ontologies even bring about Hasse diagrams with a depth of 10 or more. The numbers for the maximal branching factor are comparable.

A long path in a Hasse diagram for regularities of class frames means that corresponding modelling structures for class frames are based on the same constituent components since $\lesssim_G$ is defined in terms of a subset relation for multisets. A large branching factor, on the one hand, means that many class frames share a common substructure, namely the modelling structure of their parent. But, on the other hand, it also means that siblings of that parent vary in terms of the modelling structures.

Similarly, a long path in a Hasse diagram for regularities for axioms (as in the case of many rich ontologies) means that many regularities are based on the same nesting of OWL constructors. And a large branching factor signifies that there is a good amount of variablitiy in term of the nesting of OWL constructors on some nesting level.

6 Related Work & Discussion

While there are many surveys of properties of existing ontologies, e.g., [4,13,23,24], there is only little research on the topic of discovering ontology patterns or reverse-engineering an ontology's design. However, two approaches in this direction are motivated on similar grounds to the ones put forward in this work.

The first approach is based on agglomerative clustering to identify commonalities for named entities in an ontology based on similar syntactic representations [15]. Similarities between these representations are distilled in the form of sets of axioms with variables. While these representations bear some similarities to the notion of modelling structures in the context of this work, there are subtle differences with regards to the underlying notion of regularity. The approach using agglomerative clustering identifies regularities for named entities, whereas the approach based on language abstractions identifies regularities for axioms (or sets of axioms). So, the former approach is primarily concerned with regularities for elements of an ontology's domain-specific vocabulary, whereas the latter focuses on regularities for syntactic structures based on an ontology's underlying formal language, e.g, OWL.

The second approach is based on frequent subtree mining over OWL axioms [12]. By interpreting OWL axioms as syntax trees, well-known subtree mining algorithms can be used to identify frequent tree structures. Furthermore, a notion for regularities for class

Table 4: Maximal nesting depth of modelling structures (left-hand side) and maximal number of axioms in class frames (right-hand side).

Max Depth	Atomic	$\mathcal{EL}^{++}$	Rich
1	94	16	107
2	-	71	116
3	-	1	83
4	-	1	36
5–9	-	1	118
≥ 10	-	-	13

Max CF Axioms	Atomic	$\mathcal{EL}^{++}$	Rich
1	54	7	43
2	21	14	46
3	6	11	49
4–9	13	35	174
10–19	-	5	73
≥ 20	-	18	88

Table 5: Depth and maximal branching factor of Hasse diagrams for posets.

Depth	Axioms			Class Frames		
	Atomic	$\mathcal{EL}^{++}$	Rich	Atomic	$\mathcal{EL}^{++}$	Rich
1	94	21	96	54	8	47
2	-	67	136	21	14	53
3	-	2	56	9	15	71
4–9	-	-	139	10	33	212
10–19	-	-	41	-	17	61
≥ 20	-	-	5	-	3	29

Branching	Axioms			Class Frames		
	Atomic	$\mathcal{EL}^{++}$	Rich	Atomic	$\mathcal{EL}^{++}$	Rich
0	94	21	96	54	8	47
1	-	66	123	40	34	64
2	-	2	55	-	43	51
3–9	-	1	174	-	5	179
10–19	-	-	25	-	-	70
≥ 20	-	-	-	-	-	62

frames is motivated that is based on identified regularities for syntax trees of axioms. For example, regularities for subsumption axioms with the same and non-variable left-hand side are grouped into a set to give rise to a new regularity for sets. In cases where the left-hand side is a variable, frequent itemset mining is proposed to identify co-occurring axioms as regularities for class frames. While the approach based on frequent subtree mining bears a resemblance to the approach based language abstractions, there are both technical differences as well as conceptual differences.

First and foremost, it is important to recognise that frequent subtree mining aims at identify regularities based on some notion of *frequency*. A tree structure is considered frequent if it satisfies some threshold criterion. However, regularities based on language abstractions are *independent* of any notion of frequency; or any other notion depending on a threshold for that matter. The importance of this needs to be emphasised because regularities based on thresholds are generally not suitable for analysing an ontology's design as a whole. The simple reason for this is that such notions, by definition, do not account for structures that do not satisfy the threshold criterion. For example, variations in the reuse of a single pattern in an ontology's design may give rise to many slightly different syntactic structures. If none of the variant reuses of the pattern gives rise to frequent structures, then no regularity (based on frequency) is identified.

In any case, any conclusion or claim about an ontology's underlying design based on syntactic regularities has to be made with due diligence regardless of the used approach. Consider for example the case of a pattern-based ontology design. A *pattern* in the context of ontology engineering often denotes a rather distinctive notion. An example of this are *Ontology Design Patterns* (ODP) that are proposed as well-proven modelling solution to common modelling problems and often provide a reusable component such as a set of axioms [3,2]. While such a reusable component is often associated with a syntactic structure, e.g., a set of axioms, the converse is not necessarily the case. Meaning, a reusable component of a pattern cannot be equated with the pattern itself and the presence of axioms associated with a pattern's reusable component cannot be equated with an actual reuse of the pattern. So, even though the discovery of regularities can be helpful to detect structures that are indicative of an ODP's reuse, a domain expert's assessment of an identified regularity in an ontology is required to gauge whether the regularity is connected to an ODP.

Even though the idea of reusable components has been popularised by the ODP community, there is no standard mechanism or de facto practice for reusing a given ODP. Despite the development of frameworks and tool support for ODPs reuse [8,17,22,25],

little is known about what kind of features are needed to facilitate pattern-based ontology engineering in practice [6]. Developing an understanding of compositional aspects of syntactic structures in ontologies w.r.t. syntactic abstractions may provide a way of informing and evaluating the design of tools and frameworks in this direction.

As an example, consider the Galen Ontology [20] in which the classes CurrentBloodPressureLevel and RecentBloodPressureLevel are represented via almost identical $EquivalentClasses$ axioms. Both use the following expression (written in infix notation):

$$\mathsf{LevelState} \sqcap (\exists \mathsf{isSpecificAnswerOf}.(\mathsf{InvestigationAct} \sqcap (\exists \mathsf{hasTimeOfOccurrence}.(\mathsf{TimeOfOccurrence} \sqcap (\exists \mathsf{hasAbsoluteState}\ atTime))) \sqcap (\exists \mathsf{isToDetermine}.\mathsf{BloodPressure})))$$

where the variable $atTime$ is set to Now and RecentPast respectively. Here, the use of the variable $atTime$ can be seen as an abstraction over differences between the representations of CurrentBloodPressureLevel and RecentBloodPressureLevel. In this case, a simple templating mechanism allowing for the *instantiation of parametrised representations*, e.g. $\mathsf{CurrentBloodPressureLevel} \equiv \mathtt{BloodPressureLevel}(\mathsf{Now})$, would be suitable to capture this abstract structure in an arguably meaningful way. So, research into the discovery of meaningful abstractions as well as suitable ways of encoding them promises to have a great impact on pattern-based ontology engineering.

7 Conclusion

In this paper, we adapted and extended a formal framework for analysing syntactic regularities in ontologies originally proposed in [9,10]. The framework is based on a syntax-directed approach that decomposes an ontology into equivalence classes of syntactic structures, where two syntactic structures are considered equivalent if they are indistinguishable under a formal notion of abstraction. We proposed the notion of a modelling structure for the purpose of analysing and characterising syntactic regularities. Furthermore, we proposed formal relations between such modelling structures so that they can be organised in terms of a partial order that captures a notion of substructure containment. Finally, we used these notions to conduct a large-scale empirical investigation of syntactic modelling structures in biomedical ontologies.

We find that most ontologies contain primarily axioms of a simple syntactic structure. However, such axioms seem to be combined in various ways to give rise to comparatively many modelling structures for class frames. This suggests that class frames play a crucial role in the representation of many entities in the biomedical domain.

Our findings on common modelling structures across biomedical ontologies reveal that only comparatively simple syntactic structures for both axioms and class frames reoccur. However, the results obtained on the maximal size and depth of modelling structures indicate that many rich ontologies also contain highly complex modelling structures in which OWL constructors are deeply nested and combined. Moreover, such complex structures are also highly interrelated w.r.t. shared substructures in many ontologies. While our investigation provides proof of structural complexities in ontologies, further research is needed to qualify underlying design rationales.

Supplemental Material Statement: Source code is available at `https://github.com/ckindermann/iswc-2022`.

References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
2. Blomqvist, E., Sandkuhl, K.: Patterns in Ontology Engineering: Classification of Ontology Patterns. In: ICEIS (3). pp. 413–416 (2005)
3. Gangemi, A.: Ontology Design Patterns for Semantic Web Content. In: ISWC. Lecture Notes in Computer Science, vol. 3729, pp. 262–276. Springer (2005)
4. Glimm, B., Hogan, A., Krötzsch, M., Polleres, A.: OWL: Yet to arrive on the Web of Data? In: LDOW. CEUR Workshop Proceedings, vol. 937. CEUR-WS.org (2012)
5. Grau, B.C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P.F., Sattler, U.: OWL 2: The next step for OWL. J. Web Semant. **6**(4), 309–322 (2008)
6. Hammar, K., Blomqvist, E., Carral, D., van Erp, M., Fokkens, A., Gangemi, A., van Hage, W.R., Hitzler, P., Janowicz, K., Karima, N., Krisnadhi, A., Narock, T., Segers, R., Solanki, M., Svátek, V.: Collected Research Questions Concerning Ontology Design Patterns. In: Ontology Engineering with Ontology Design Patterns, Studies on the Semantic Web, vol. 25, pp. 189–198. IOS Press (2016)
7. Horridge, M., Patel-Schneider, P.F.: Manchester Syntax for OWL 1.1. In: OWLED (Spring). CEUR Workshop Proceedings, vol. 496. CEUR-WS.org (2008)
8. Iannone, L., Rector, A.L., Stevens, R.: Embedding Knowledge Patterns into OWL. In: ESWC. Lecture Notes in Computer Science, vol. 5554, pp. 218–232. Springer (2009)
9. Kindermann, C.: Analysing Patterns and Regularities in Ontologies. Ph.D. thesis, University of Manchester, UK (2022)
10. Kindermann, C., Parsia, B., Sattler, U.: Syntactic Regularities Based on Language Abstractions. Advances in Pattern-Based Ontology Engineering **51**, 312 (2021)
11. Kindermann, C., Skjæveland, M.G.: A Survey of Syntactic Modelling Structures in Biomedical Ontologies. CoRR **abs/2207.14119** (2022), https://arxiv.org/abs/2207.14119
12. Lawrynowicz, A., Potoniec, J., Robaczyk, M., Tudorache, T.: Discovery of emerging design patterns in ontologies using tree mining. Semantic Web **9**(4), 517–544 (2018). https://doi.org/10.3233/SW-170280, https://doi.org/10.3233/SW-170280
13. Matentzoglu, N., Bail, S., Parsia, B.: A Snapshot of the OWL Web. In: ISWC (1). Lecture Notes in Computer Science, vol. 8218, pp. 331–346. Springer (2013)
14. Matentzoglu, N., Parsia, B.: BioPortal Snapshot 30.03.2017 (Mar 2017). https://doi.org/10.5281/zenodo.439510, https://doi.org/10.5281/zenodo.439510
15. Mikroyannidi, E., Iannone, L., Stevens, R., Rector, A.L.: Inspecting Regularities in Ontology Design Using Clustering. In: ISWC (1). Lecture Notes in Computer Science, vol. 7031, pp. 438–453. Springer (2011)
16. Motik, B., Patel-Schneider, P., Bock, C., Fokoue, A., Haase, P., Hoekstra, R., Horrocks, I., Ruttenberg, A., Sattler, U., Smith, M.: OWL 2 Web Ontology Language: Structural Specification and Functional-Style (01 2008), WC3 Recommendation
17. Noppens, O., Liebig, T.: Ontology Patterns and Beyond - Towards a Universal Pattern Language. In: WOP. CEUR Workshop Proceedings, vol. 516. CEUR-WS.org (2009)
18. Noy, N.F., Musen, M.A., Jr., J.L.V.M., Rosse, C.: Pushing the envelope: challenges in a frame-based representation of human anatomy. Data Knowl. Eng. **48**(3), 335–359 (2004)
19. Osumi-Sutherland, D., Courtot, M., Balhoff, J.P., Mungall, C.J.: Dead simple OWL design patterns. J. Biomed. Semant. **8**(1), 18:1–18:7 (2017)

20. Rector, A.L., Rogers, J.: Ontological and Practical Issues in Using a Description Logic to Represent Medical Concept Systems: Experience from GALEN. In: Reasoning Web. Lecture Notes in Computer Science, vol. 4126, pp. 197–231. Springer (2006)
21. Sarntivijai, S., Lin, Y., Xiang, Z., Meehan, T.F., Diehl, A.D., Vempati, U.D., Schürer, S.C., Pang, C., Malone, J., Parkinson, H.E., Liu, Y., Takatsuki, T., Saijo, K., Masuya, H., Nakamura, Y., Brush, M.H., Haendel, M.A., Zheng, J., Jr., C.J.S., Peters, B., Mungall, C.J., Carey, T.E., States, D.J., Athey, B.D., He, Y.: CLO: the cell line ontology. J. Biomed. Semant. **5**, 37 (2014)
22. Skjæveland, M.G., Lupp, D.P., Karlsen, L.H., Forssell, H.: Practical Ontology Pattern Instantiation, Discovery, and Maintenance with Reasonable Ontology Templates. In: ISWC (1). Lecture Notes in Computer Science, vol. 11136, pp. 477–494. Springer (2018)
23. Sváb-Zamazal, O., Svátek, V.: Analysing Ontological Structures through Name Pattern Tracking. In: EKAW. Lecture Notes in Computer Science, vol. 5268, pp. 213–228. Springer (2008)
24. Wang, T.D., Parsia, B., Hendler, J.A.: A Survey of the Web Ontology Landscape. In: ISWC. Lecture Notes in Computer Science, vol. 4273, pp. 682–694. Springer (2006)
25. Warrender, J.D., Lord, P.: A Pattern-driven Approach to Biomedical Ontology Engineering. In: SWAT4LS. CEUR Workshop Proceedings, vol. 1114. CEUR-WS.org (2013)