View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Ghent University Academic Bibliography

MontoloStats - Ontology Modeling Statistics

Sven Lieber
sven.lieber@ugent.be
Ghent University – imec – IDLab, Department of Electronics and Information Systems
Ghent, Belgium

Ben De Meester
ben.demeester@ugent.be
Ghent University – imec – IDLab, Department of Electronics and Information Systems
Ghent, Belgium

Anastasia Dimou
anastasia.dimou@ugent.be
Ghent University – imec – IDLab, Department of Electronics and Information Systems
Ghent, Belgium

Ruben Verborgh
ruben.verborgh@ugent.be
Ghent University – imec – IDLab, Department of Electronics and Information Systems
Ghent, Belgium

ABSTRACT

Within ontology engineering concepts are modeled as classes and relationships, and restrictions as axioms. Reusing ontologies requires assessing if existing ontologies are suited for an application scenario. Different scenarios not only influence concept modeling, but also the use of different restriction types, such as subclass relationships or disjointness between concepts. However, metadata about the use of such restriction types is currently unavailable, preventing accurate assessments for reuse. We created the RDF Data Cube-based dataset *MontoloStats*, which contains restriction use statistics for 660 LOV and 565 BioPortal ontologies. We analyze the dataset and discuss the findings and their implications for ontology reuse. The *MontoloStats* dataset reveals that 94% of *LOV* and 95% of *BioPortal* ontologies use RDFS-based restriction types, 49% of *LOV* and 52% of *BioPortal* ontologies use at least one OWL-based restriction type, and different literal value-related restriction types are not or barely used. Our dataset provides modeling insights, beneficial for ontology reuse to discover and compare reuse candidates, but can also be the basis of new research that investigates novel ontology engineering methodologies with respect to restrictions definition.

KEYWORDS

ontology engineering, restrictions, axioms, statistics, RDF

ACM Reference Format:
Sven Lieber, Ben De Meester, Anastasia Dimou, and Ruben Verborgh. 2019. MontoloStats - Ontology Modeling Statistics. In *10th International Conference on Knowledge Capture (K-CAP'19), November 19–21, 2019, Marina Del Rey, CA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3360901.3364433

1 INTRODUCTION

The Semantic Web uses *ontologies* to formally represent real-world domains and concepts [17]. An ontology is a conceptualization, an intensional semantic structure which encodes the implicit rules restricting the structure of a piece of reality [6]. In addition to containing *concepts* and *relationships*, an ontology is characterized by a set of *axioms* [5]. We consider everything expressible as axiom a *restriction*. Different types of restrictions exist, such as subclass relationships or disjointness between concepts. Each restriction type serves different purposes: subclass relationships can for instance describe taxonomic structures, and disjoint classes express mutual exclusiveness in a machine-understandable way.

Ontologies play different roles in different application scenarios [16], influencing how restrictions are used. In a semantic search scenario, ontologies are built to be used by machines, which demands *machine-understandable semantics* that are explicitly stated restrictions, such as cardinalities or disjoint properties. In more human-targeted scenarios, such heavily axiomatized ontologies would pose challenges regarding comprehensibility. For instance, a taxonomic structure defined with restriction type *subsumption*, when encoded using a `rdfs:subClassOf` expression, imposes lower ontology reusability costs than other restriction types [18].

Whereas several insights on class and relationship usage exist, restriction types so far have remained insufficiently documented, making it difficult to inform ontology reuse. From a process point-of-view ontology reuse consists of multiple activities, such as *discovery* and *assessment* of reuse candidates [16]. Metadata about prevalent restriction types would support the selection of reuse candidates that are appropriate for a given application scenario. Restriction types can be expressed with different vocabularies and terms, and, thus, multiple expressions need to be considered to obtain comprehensive metadata. Consider for instance disjoint class restrictions which can be expressed either using the property `owl:disjointWith`, or the class `owl:AllDisjointClasses`.

To the best of our knowledge, currently no available dataset exists which provides statistics about restriction type use independent from their expressions.

We introduce the *MontoloStats* dataset describing the use of different restriction type expressions in *LOV* and *BioPortal* ontologies. We analyze the dataset and discuss the results with respect to ontology reuse. Our contributions are:

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

K-CAP '19, November 19–21, 2019, Marina Del Rey, CA, USA

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-7008-0/19/11…$15.00
https://doi.org/10.1145/3360901.3364433

(1) an approach to model restriction types' expressions and statistical measures using the W3C-recommended RDF Data Cube and PROV vocabularies;
(2) an implementation of the approach as extension of *LODStats* [2] to automatically generate statistical measures;
(3) the *MontoloStats* statistical dataset to describe restrictions use in *LOV* and *BioPortal* ontologies;
(4) analysis and discussion of the *MontoloStats* dataset and its implications for ontology reuse and further research.

MontoloStats can foster further research in a plethora of research challenges related to, for instance, ontology reuse to assess different aspects of an ontology, or knowledge modeling. Statistics regarding the restriction type use and distribution may be used for further in-depth analysis of how restriction types were modeled as axioms and what impact this has on their further use. The resources accompanying this paper are published at https://w3id.org/montolo, specifically:

- The **MontoloStats statistical dataset** is published at https://w3id.org/montolo/data/montolo-stats/ under CC0 license[1], with accompanying public SPARQL endpoint (DOI: 10.5281/zenodo.3407139);
- Definitions of identified restriction types, expressions and measures are published as **Montolo dataset** at https://w3id.org/montolo/ns/montolo under CC0 license (DOI: 10.5281/zenodo.3343313);
- The **MontoloVoc vocabulary**, created to describe concepts of *Montolo* and *MontoloStats* is published at https://w3id.org/montolo/ns/montolo-voc and made available at https://github.com/IDLabResearch/montolo-voc under CC0 license (DOI: 10.5281/zenodo.3343335);
- The **LODStats extension** used to create *MontoloStats* is available at https://github.com/IDLabResearch/lovstats under MIT license[2] (DOI: 10.5281/zenodo.2165747).

The remainder of the paper is organized as follows: Section 2 summarizes the related work, in Section 3 we present our proposed approach, that generates *MontoloStats* (Section 4). Last, we analyze *MontoloStats* in Section 5 and summarize our conclusions and future work in Section 6.

2 RELATED WORK

Our work concerns statistics regarding the use of restrictions to support ontology reuse. Therefore, we investigate existing work regarding (i) restrictions in ontologies, (ii) ontology reuse, and (iii) statistics in the Semantic Web.

Restrictions. More complex and possibly formal vocabularies containing restrictions, are usually referred to as ontologies[3] and aim to represent knowledge machine-understandably. OWL2 is a knowledge representation language which uses different restriction types in the form of axioms, e.g. *disjoint classes* or *reflexive properties.* Whereas restrictions in the form of axioms are used to represent knowledge, restrictions in the form of constraints are used to e.g., validate data which should adhere to such a knowledge representation [9].

The latter was investigated mostly in the context of data quality. RDFUnit [8] is a test-driven evaluation framework for Linked Data, which uses a set of SPARQL templates, expressing data quality issues. Several Data Quality Test Patterns cover aspects such as cardinality, disjointness or literal value restrictions.

Arndt et al. [1] provided an alignment between RDFUnit's Data Quality Test Patterns and corresponding restriction types identified by Hartmann [7], to cover restriction types which minimally cover common validation requirements. An investigation in the use of such restriction types in ontologies could reveal beneficial information for ontology engineering.

Ontology Reuse. Ontology reuse implicitly follows a four step workflow involving the discovery, selection, customization and integration of potential reuse candidates [15]. Different methods exist to support each step's tasks, and especially ontology metadata can be of use for the first two steps.

The first step, discovery of existing ontologies and their concepts, is facilitated by vocabulary catalogs such as LOV [20] or Bioportal [12]. These catalogues provide search capabilities already considering a limited amount of metadata.

However, given an application scenario in which more or less axiomatized ontologies are required, the current search capabilities are insufficient, i.e. no filter on ontologies using specific restriction types or restriction type expressions. These search capabilities, and hence the ontology discovery step, would benefit from restriction use statistics.

The second step, selection of appropriate reuse candidates, entails the evaluation of the different reuse candidates with respect to the given application scenario.

OOPS! [12] validates ontologies by detecting anomalies and bad practices leading to modeling errors, thus, it supports users to qualitatively evaluate and compare reuse candidates.

Our statistics provide quantitative measurements of restriction type use which can complement a qualitative assessment and support users in selecting ontologies appropriate for given application scenarios with respect to modeled restrictions.

From an economical point-of-view the activities performed in a reuse process adhere to different costs. The ONTOCOM [18] cost estimation model, created based on expert interviews [17], tries to quantify these costs by calculating necessary person-months effort.

Several identified cost drivers could benefit from restriction use statistics, as users' effort may be reduced due to available restriction use statistics for ontology reuse related tasks.

Statistics in the Semantic Web. Two main approaches to compute statistics were suggested: from a dataset and from an ontology point-of-view. Datasets are statistically analyzed in RDFStats [10], LODStats [2] and Loupe [11]. RDFStats [10] supports users to browse RDF graphs and applications dealing with large, possibly distributed RDF graphs. Statistical metrics of RDFStats were reused in LODStats [2], a statement-stream-based approach to analyze RDF data. LODStats, due to its streaming approach, is suitable for large datasets. It comes with a set of 32 statistical measures, which can be extended. Loupe [11], among others, analyzes implicit data patterns, regarding

[1] https://creativecommons.org/publicdomain/zero/1.0/
[2] https://opensource.org/licenses/MIT
[3] https://www.w3.org/standards/semanticweb/ontology

vocabulary use, and explicit vocabulary definitions regarding ontological axioms used in data. Focused on dataset structure, Loupe does not cover restriction-related information.

Dataset-related approaches focus on dataset structure, schema-level statistics are only considered to a small extent. Additionally, restrictions are covered from a dataset point-of-view, creating mixed statistics of all ontologies used in a dataset. Ontology reuse concerns the discovery and selection of possible reuse candidates, and if compared based on statistical metadata, restriction use statistics from an ontology point-of-view are needed.

From an ontology point-of-view, tools like Protégé [13] provide summaries about used axioms in an ontology, but these summaries only cover a fixed set of axioms, and are only shown for the currently loaded ontologies. In contrast, our approach describes generic restriction types and concrete expressions which are extendible and provides a statistical dataset covering multiple ontologies.

ComplexOnto [3] is a score, expressing the complexity of ontologies, to better understand, maintain, reuse and integrate ontologies. The score consists of four metrics describing different interlinking characteristics, based on properties and subclass axioms. However, the score, as aggregated value, does not provide detailed information, and its constituents only focus on how connected used concepts are, leaving out information regarding used axioms.

The discovery and selection of ontologies for reuse based on statistical metadata regarding restriction use demands available restriction use statistics per ontology. Additionally, vocabularies such as RDF and OWL contain different expressions for identified restriction types which need to be considered to provide comprehensive statistics. To the best of our knowledge, existing approaches do not provide statistics on restriction use per ontology on the level of restriction types taking different expressions into account. Existing approaches do, however, provide a framework to create statistics which we extend for restriction use in ontologies.

3 APPROACH

We propose an approach to compute statistics of restriction type use in ontologies to support ontology engineering activities. We differentiate between (abstract) *restriction types*, e.g. disjointness, and (concrete) *restriction type expressions* per restriction type, e.g. disjoint classes expressed via the property `owl:disjointWith` or the class `owl:AllDisjointClasses`, to comprehensively describe restriction use information. More, we define measures to calculate statistics of restriction types and their expressions, e.g. number of occurrences of classes annotated with `owl:disjointWith`. Our approach consists of three steps: (i) unambiguously description of restrictions, (ii) extraction of restriction type expressions from ontologies, and (iii) computation of statistics, described with our RDF DataCube-based *MontoloVoc* vocabulary.

1. describe restriction types, expressions and measures. We followed the UPON-light methodology [4] to create our *MontoloVoc* vocabulary describing restriction types, their expressions and measures in a machine-understandable way. Restriction types and expressions can be defined and linked using the associated *MontoloVoc* classes[4], thus measured values can be linked to a single definition. An instance of the class `mov:RestrictionType` is created for each restriction type, as shown in Listing 2, line 1-4 for the restriction type *disjoint classes*. Different expressions of this restriction type, such as `owl:disjointWith` (6-10) or `owl:AllDisjointClasses` (line 14-16) can be created using the introduced *MontoloVoc* class `mov:RestrictionTypeExpression`, which is linked via the property `frbr:realizationOf`[5] to their respective `mov:RestrictionType`, to make their relationship explicit. Different measures can be defined to analyze restriction use in ontologies. A measure, e.g. number of occurrences, can be described with the *MontoloVoc* class `mov:RestrictionTypeMeasure` (line 20-21).

2. extract restriction type expressions from ontologies. This step concerns the extraction of identified restriction type expressions from ontologies. Different extraction mechanisms can be used for this step, e.g. queries on ontologies or stream-based solutions reading RDF.

```
1   # instances of collection cannot be instances of
2   # concepts or concept schemes and vice versa
3   skos:Collection
4     owl:disjointWith skos:Concept ;
5     owl:disjointWith skos:ConceptScheme .
6
7   skos:ConceptScheme owl:disjointWith skos:Concept .
8
9   # same restriction expressed using a class (pairwise exclusive)
10  [] a owl:AllDisjointClasses ;
11  owl:members ( skos:Collection skos:ConceptScheme skos:Concept).
```

Listing 1: Disjoint classes restriction, expressed with OWL in 2 different semantically equivalent ways.

3. compute restriction type measures. Different measures can be defined to analyze restriction type use in ontologies, but need to be computed differently for each restriction type expression. Measures relate to restriction types, but to achieve a fair comparison between different restriction type expressions, the measure needs to be computed differently. Consider again the restriction type *disjoint classes*. The three RDF statements in Listing 1 line 3-7 express the disjointness between `skos:Collection`, `skos:Concept` and `skos:ConceptScheme`, and therefore correspond to three restriction statements. Yet the two RDF statements in Listing 1 line 10-11 also define three restrictions. An OWL restriction class instance with a list of pairwise disjoint classes is used, which corresponds to $\frac{n^2-n}{2}$ disjoint class statements. Both expressions lead to three disjoint classes, although the number of RDF statements differs. Hence the number of *disjoint classes* restrictions need to be computed differently for each expression, to achieve comparable restriction type measures between restriction type expressions. The computed measures can then be described with the class `lst:RestrictionTypeStatistic` (Listing 2 line 23-33), subclass of an RDF data cube observation.

4 MONTOLO

We applied the approach to create both *Montolo*, descriptions of restriction types, and *MontoloStats*, a dataset describing restriction type use of LOV and Bioportal ontologies. In the following we

[4] Abbreviated in this paper using the mov prefix.

[5] http://purl.org/vocab/frbr/core#

```
# Restriction Type
mon:disjointClasses
  a mov:RestrictionType ;
  rdfs:label "Disjoint classes restriction type"@en .

# Restriction Type Expression 1
mon:disjointClassesOwlDisjointWith
  a mov:RestrictionTypeExpression ;
  frbr:realizationOf mon:disjointClasses ;
  rdfs:label "owl:disjointWith restriction"@en .

# Restriction Type Expression 2
mon:disjointClassesOwlAllDisjointClasses
  a mov:RestrictioinTypeExpression ;
  frbr:realizationOf mon:disjointClasses ;
    rdfs:label "owl:AllDisjointClasses restriction"@en .

# Restriction Type Measure
mon:restrictionTypeOccurrence
  a mov:RestrictionTypeMeasure ;
  rdfs:label "Restriction type occurrence"@en .

# Restriction Type Statistic (example of a generated result)
[] a mov:RestrictionTypeStatistic ;
  mon:executionTimeDimension
    "2019-04-06T08:30:54.280117"^^xsd:dateTime ;
  mon:detectorVersionDimension
    mon:disjointClassesLODStatsDetectorOwlDisjointWith-v1 ;
  mon:ontologyRepository mon:lov ;
  mon:ontologyVersionDimension
    <http://www.w3.org/2004/02/skos/core#> ;
  mon:restrictionTypeDimension mon:disjointClasses ;
  mon:restrictionTypeOccurrence 3 .
```

Listing 2: Restriction type *disjoint classes* and its expressions in Montolo namespace (prefix `mon`), represented with *MontoloVoc* vocabulary (prefix `mov`).

describe (i) restriction types we cover in *Montolo*, (ii) the implementation of our approach as LODStats extension, and (iii) the *MontoloStats* dataset.

4.1 Covered restriction types and measures

We described 18 restriction types based on related work [1, 7], using the proposed *MontoloVoc* vocabulary. We also define the *occurrence* measure expressing the number of axiom statements, following step 1 of our approach[6]. Table 1 lists the restriction types and restriction type expressions used to detect them. We consider restriction types expressed using RDFS and OWL vocabularies, because dataset-related statistics indicate that RDF(S) and OWL are the most prevalent vocabularies to define ontologies using RDF [2, 19].

From RDFS, we cover the three restriction types *subsumption*, *domain* and *range* to identify taxonomic structures. For the expression `rdfs:subClassOf` we also use the `isIRI` filter provided by LODStats to count actual taxonomic relationships between concepts and avoid counting common patterns in which e.g. `rdfs:subClassOf` is used to express that a concept is a subclass of a specific `owl:Restriction`. Furthermore we consider all six *cardinality-related restriction types* that OWL describes. For the restriction type *exact unqualified cardinality*, we cover two expressions: the property `owl:cardinality` and a combination of `owl:minCardinality` and `owl:maxCardinality` with the same value. Also two expressions are defined for each of the two restriction types *disjoint classes* and *disjoint properties*, as machine-understandable disjointness is an important information for the Semantic Web. We also consider different *property* and *literal value-related* restriction types.

4.2 LODStats extension

We build upon and contribute to existing work to provide statistics about restriction types. We take advantage of *LODStats*' extensibility to define statistical modules for restriction types. For each restriction type, we create one statistical module. Restriction types can be expressed in different ways, yet restriction type measures should be comparable between restriction type expressions. Thus, we introduce one *detector class* per restriction type expression which shares the same interface among its corresponding restriction type and provides same measures. Other restriction types can be added as statistical modules and other restriction type expressions can be added using a new detector. Thus our implementation adheres to the extendibility of our approach.

4.3 Dataset

We applied the approach on two repositories: (i) *LOV*, a general-purpose ontology repository, and (ii) *BioPortal*, a domain-specific ontology repository. The *MontoloStats* dataset consists of 395,675 triples and 31,850 RDF data cube observations. The *MontoloStats* dataset is small in size (22 MB) and interoperable as it adheres to the W3C recommendations RDF DataCube and PROV. We published *MontoloStats* on Zenodo under CC0 license to ensure its availability. All Montolo-related artifacts, such as the *MontoloVoc* vocabulary and LODStats extension, are publicly hosted on GitHub, to enable the community's engagement.

We provide badges for each ontology indicating the number of prevalent restriction types. Such badges allows for easy visual inspection and comparison of vocabularies, and eases integration in existing platforms and systems. Badges are available for every ontology in the *MontoloStats* dataset[7], redirecting to the detailed *MontoloStats* page per ontology[8].

LOV. We analyzed ontologies listed in *LOV*, which contained by the time of writing 672 ontologies. We downloaded the latest version of each ontology from *LOV* in N-triples and stored them. Due to some errors during parsing, we could compute our statistics for 660 ontologies and, thus, the statistics cover 98% of *LOV*.

BioPortal. We analyzed OWL and OBO ontologies, which are OWL-compatible, listed in *BioPortal*. According to a JSON file obtained via BioPortal[9], 716 OWL and 123 OBO ontologies are listed. However, while downloading the ontologies we encountered several *Access Denied* responses due to a missing ontology file or license-restrictions. We used the *robot* tool [14] to convert the downloaded OWL/XML and OBO ontologies to RDF/XML, as it adheres to the W3C recommended OWL-TO-RDF mapping[10] and supports the OBO format. The conversion failed for 87 ontologies due to different parsing errors. Finally, the successful converted ontologies were transformed to N-triples and, thus, we could compute our statistics for 565 ontologies of BioPortal.

[6] https://w3id.org/montolo/ns/montolo

[7] https://w3id.org/montolo/data/montolo-stats/latest/voc/[prefix]?type=svg.

[8] https://w3id.org/montolo/data/montolo-stats/latest/voc/[prefix].

[9] http://data.bioontology.org/ontologies_full

[10] https://www.w3.org/TR/owl2-mapping-to-rdf/

Table 1: Restriction types and corresponding expressions to detect them. Restriction type expressions are listed as triple patterns and additional filter functions. For each found triple pattern we increase the corresponding counter by 1, except for the 2nd expression of *disjoint classes* and *properties*, where we compute $\frac{n^2-n}{2}$ (n is the $?list$'s length).

Restriction Type	Restriction Type Expression
Subsumption	`{?s rdfs:subClassOf ?o .}` *&& isIRI(?s) && isIRI(?o)*
Domain	`{?s rdfs:domain ?o .}`
Range	`{?s rdfs:range ?o .}`

Literal pattern matching	`{?s owl:withRestrictions ?list .`
	`?s2 xsd:pattern ?o2 . }`
	isListMember(?list, ?s2)

Literal ranges	`{?s owl:withRestrictions ?list .`
	`?s2 xsd:minInclusive|xsd:minExclusive`
	`|xsd:maxInclusive|xsd:maxExclusive ?o2 .}`
	`&& isListMember(?list, ?s2)`

Min unqualified cardinality	`{?s owl:minCardinality ?o .}`
Min qualified cardinality	`{?s owl:minQualifiedCardinality ?o .}`
Max unqualified cardinality	`{?s owl:maxCardinality ?o .}`
Max qualified cardinality	`{?s owl:maxUnqualifiedCardinality ?o .}`
Exact qualified cardinality	`{?s owl:qualifiedCardinality ?o .}`

Exact unqualified cardinality	`{?s owl:cardinality ?o .}`
	`{?s1 owl:minCardinality ?o1 .`
	`?s2 owl:maxCardinality ?o2 .}`
	`&& isEqual(?o1, ?o2)`

Functional properties	`{?s rdf:type owl:FunctionalProperty .}`
Inverse functional properties	`{?s rdf:type owl:InverseFunctionalProperty.}`
Universal quantification	`{?s owl:allValuesFrom ?o .}`
Asymmetric properties	`{?s rdf:type owl:AsymmetricProperty .}`
Irreflexive properties	`{?s rdf:type owl:IrreflexiveProperty .}`

Disjoint properties	`{?s owl:propertyDisjointWith ?o .}`
	`{?s rdf:type owl:AllDisjointProperties .`
	`?s owl:members ?list .} && isEqual(?o1, ?o2)`

Disjoint classes	`{?s owl:disjointWith ?o .}`
	`{?s rdf:type owl:AllDisjointClasses .`
	`?s owl:members ?list .} && isEqual(?o1, ?o2)`

5 ANALYSIS

We analyze the restriction type distribution to provide an overview of their use in *LOV* and *BioPortal* and multiple expressions for restriction types to reveal modeling practices.

5.1 Restriction Type Distribution

We analyze *MontoloStats* with respect to (i) the distribution of restriction types across *LOV* and *BioPortal*, (ii) vocabularies used to encode restriction type expressions, (iii) cardinality-related and (iv) property-related restriction types, and (v) ontologies using no restriction types.

Restriction Types. In total, 17 out of 18 restriction types occur in both LOV and BioPortal ontologies, from which 15 barely appear and 3 clearly dominate in LOV (Figure 1), and only 1 in BioPortal. 3 restriction types, namely *subsumption*, *domain*, and *range* in its RDFS-based expressions `rdfs:subClassOf`, `rdfs:domain` and `rdfs:range` stand out in LOV, as each of them occurs more than 27,000 times in total and in more than 94% of LOV ontologies. This indicates a taxonomic structure of the ontological concepts for the majority of LOV ontologies. Similarly, *subsumption* is also the most used restriction type in BioPortal, occurring more than 3 million times in total and in more than 93% of BioPortal ontologies. The restriction types *domain* and *range* are not as common in BioPortal as they are in LOV, both total numbers and amount of ontologies using it is considerably lower. But therefore *disjoint classes* restrictions are the second most used restrictions in BioPortal, used more than 760,000 times and in around 38% of the analyzed BioPortal ontologies. By total number, *subsumption* is the most used restriction type in both LOV and BioPortal ontologies. The restriction type *range* is the most used in 88% of LOV ontologies, and *subsumption* restrictions are the most used restrictions in BioPortal ontologies with 93%.

On the other end of the spectrum, the restriction type *literal ranges* occurs only 64 times in 4 LOV ontologies, and 421 times in 13 BioPortal ontologies. This corresponds to less than 1% of the LOV and around 2% of BioPortal ontologies. Neither LOV nor BioPortal ontologies have the restriction type *literal pattern matching*. We assume that restrictions regarding literal values are either not popular, or the ontologies are modeled in such a way, that literal values-related restrictions are not necessary (a concept expressed as class rather than literal value). Whereas the restriction type *literal ranges* is the least used in LOV ontologies, for BioPortal it is the restriction type *asymmetric properties*.

For BioPortal, trends in the total number of *subsumption* and *disjoint classes* are different compared to the number of ontologies using these restriction types. A few ontologies make heavy use of these restriction types and thus distort the result. This is different in LOV ontologies where for the 5 most-common restriction types the trends are similar between the total occurrence of a restriction type and ontologies using it, i.e. *subsumption*, *domain* and *range* dominate followed by *disjoint classes* and *universal quantification*.

Vocabularies used to express restriction types. *MontoloStats* contains information about restriction types expressed with RDFS and OWL, for which LOV and BioPortal show similar use. More than 94% of both LOV and BioPortal ontologies include at least one of the RDFS-based restrictions *subsumption*, *domain* or *range*. OWL-based restrictions are used less than RDF-based restrictions, but again to a similar extent among LOV and BioPortal with 49% respectively 52% of ontologies using it. Considered individually, the OWL-based restriction types are used in less than 26% of ontologies in both LOV and BioPortal ontologies.

Cardinality-related restriction types. Six restriction types regarding cardinality exist in *Montolo*: minimum and maximum qualified and unqualified cardinality, and exact qualified and unqualified cardinality. *MontoloStats* reveals a similar amount of use, but different use patterns between LOV and BioPortal ontologies.

In total, 24% of LOV and 21% of BioPortal ontologies use at least 1 of the 6 cardinality-related restriction types, demonstrating similar cardinality-related restriction type use in LOV and BioPortal ontologies. The *exact unqualified cardinality* restriction type is used 1,378 times in 110 ontologies, which corresponds to 16% of LOV ontologies, and, thus, the most used cardinality-related restriction type. In BioPortal ontologies, however, *minimum qualified cardinality* is the most used cardinality-related restriction type, used 1,166 times in 82 ontologies (14% of BioPortal ontologies). Comparing qualified and unqualified variants, *MontoloStats* reveals that unqualified variants are used more often than qualified in LOV ontologies, but qualified variants for *maximum* and *minimum* are more often used for BioPortal ontologies.

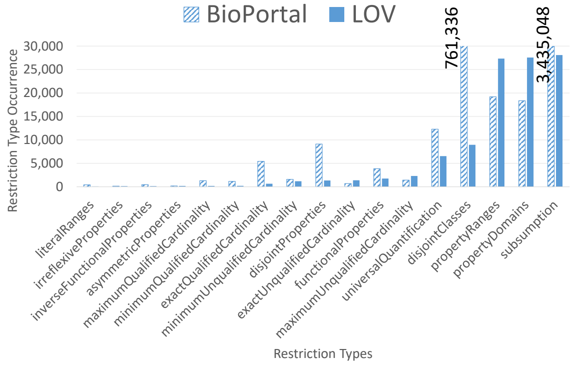

Figure 1: In 660 LOV ontologies, 3 restriction types were very common; the others were barely used. And across all 565 BioPortal ontologies, *subsumption* restrictions clearly dominate, followed by *disjoint classes* restrictions; their total occurrence is indicated as it is out of the chart bounds.

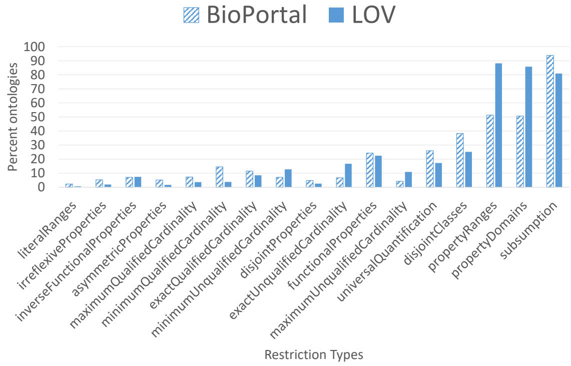

Figure 2: Restriction type use pattern is similar for LOV and BioPortal; less common OWL-based restrictions are used slightly more often in BioPortal.

In LOV ontologies, the unqualified variant of *maximum cardinality* restrictions is used 12 times more than the qualified, for *minimum cardinality* the unqualified variant is used 6 times more, and for *exact cardinality* the unqualified variant is used 2 times more. Besides these total numbers, in all 3 cases the unqualified variant is used between 2 (*exact cardinality*) and 3.4 (*minimum cardinality*) times more ontologies. While the number of ontologies for which unqualified variants are used more often is in the same range (2, 3 and 3.4 times respectively), we clearly see a trend in total numbers (12, 6 and 2 times more often), perhaps because *qualified cardinalities* were only introduced in OWL2[11], or because *qualified cardinalities* are more specific than unqualified cardinalities, which may explain that they are used less.

Compared to the above analysis of qualified and unqualified cardinalities for LOV ontologies, BioPortal ontologies show a different use. Whereas the qualified variants for *minimum* and *maximum cardinality* restrictions are used slightly less in total numbers, they are used in 2 times more ontologies. *Exact qualified* cardinalities are almost used in 2 times more ontologies and additionally 8 times more in total numbers. Thus, qualified variants of all cardinality-based restriction types seem to be more popular for BioPortal ontologies.

Property-related restriction types. Different property-related restriction types are used in 226 LOV and 219 BioPortal ontologies, corresponding to around 34% and 38% of LOV and BioPortal ontologies respectively. However, from those restriction types only *functional properties* and *universal quantification* are used to a larger extent in 22% and 17% of LOV ontologies respectively. These 2 restriction types show similar statistics for BioPortal ontologies, with the only difference that *universal quantification* restrictions are slightly more used than *functional properties* restrictions, in 26% and 24% of BioPortal ontologies respectively. The remaining property-related restriction types are barely used by the ontologies, ranging from 2% to 7% of ontologies for both LOV and BioPortal.

Ontologies using no restriction types. We found 22 LOV and 25 BioPortal ontologies which do not contain any of our identified restriction types at all. Interestingly, the Dataset Usage Vocabulary (duv) from W3C[12], part of LOV ontologies, does contain a *subsumption* restriction type. However, their used `rdfs:subClassOf` expression is differently capitalized (`rdfs:subclassOf`), which does not comply to IRI-equality[13], and was thus not considered.

5.2 Restriction Type Expressions

Besides occurrence of restriction types, *Montolo* provides information regarding occurrence of different restriction types expressions, allowing to compare different modeling practices. We provide different restriction type expressions for the following restriction types:*disjoint classes*, *disjoint properties* and *exact unqualified cardinality.*

Disjoint Classes. The *disjoint classes* restriction type can be expressed using the single property expression `owl:disjointWith`, and the list-based expression `owl:AllDisjointWith`, for which we found that the single property expression is more popular in both LOV and BioPortal ontologies.

For the `owl:disjointWith` expression of the *disjoint classes* restriction type, we count 5,303 axiom statements in 155 LOV ontologies, and 133,738 axiom statements in 203 BioPortal ontologies. Although this expression is used in a similar number of ontologies among LOV and BioPortal, the BioPortal ontologies make significantly more use of it.

The `owl:AllDisjointWith` expression of the *disjoint classes* restriction type counts 3,642 axiom statements in 34 of LOV and 627,598 axiom statements in 85 of BioPortal ontologies.

The `owl:AllDisjointWith` expression is also used to a much larger extent by total numbers in BioPortal ontologies compared

[11] https://www.w3.org/TR/owl2-new-features/#F5:_Property_Qualified_Cardinality_Restrictions

[12] https://www.w3.org/TR/vocab-duv/

[13] https://www.w3.org/TR/rdf11-concepts/#section-IRIs

to LOV ontologies, indicating more machine-understandable disjointness which may facilitate reasoning tasks. However, only 5% of LOV and 15% of BioPortal ontologies use this expression.

Comparing the 2 different expressions for *disjoint classes* restriction type, we see differences between LOV and BioPortal. In LOV, the single property `owl:disjointWith` expression compared to the list-based `owl:AllDisjointWith` is used slightly more in total numbers, but in 4.5 times more ontologies. Similarly, in BioPortal the property-based expression compared to the list-based expression is used in 2 times more ontologies. However, in total numbers BioPortal ontologies encode 4 times more concepts using the list-based expression compared to the single property expression. This indicates that BioPortal ontologies using the list-based expression encode lots of mutual exclusive disjointness.

Disjoint Properties. The *disjoint properties* restriction type can be expressed with the property expression `owl:propertyDisjointWith`, and the list-based expression `owl:AllDisjointProperties`, for which we found that the single property expression is more popular in both LOV and BioPortal ontologies.

The `owl:propertyDisjointWith` expression is used 920 times in 17 LOV and 45 times in 21 BioPortal ontologies.

The `owl:AllDisjointProperties` expression is used 424 times in 4 LOV and 9,070 times in 6 BioPortal ontologies. The property expression `owl:propertyDisjointWith` is used in 4 times more ontologies for both LOV and BioPortal ontologies. Even if a few of LOV and BioPortal ontologies heavily use the list-based expression `owl:AllDisjointProperties`, the overall trend suggests that the single property-based expression `owl:propertyDisjointWith` is more popular.

Cardinality Restrictions. The *exact unqualified cardinality* restriction type can be expressed with the property `owl:cardinality`, and a combination of `owl:minCardinality` and `owl:maxCardinality` with the same value. The latter expression is barely or not used at all which indicates that the `owl:cardinality` expression is common practice to express *exact unqualified cardinality* in both LOV and BioPortal ontologies.

The `owl:cardinality` expression is used 1,375 times in 108 LOV and 692 times in 38 BioPortal ontologies. Compared to that, the combination of `owl:minCardinality` and `owl:maxCardinality` is used only 3 times in 2 LOV ontologies, and not used at all in BioPortal ontologies. This states the use of `owl:cardinality` is not just more popular, but common practice to express *exact unqualified cardinality* restrictions in LOV and BioPortal ontologies.

6 CONCLUSIONS

We discuss findings, *MontoloStats*' potential for ontology reuse, lessons learned, and future evaluation plans.

Findings. Even though the selected repositories cover different domains (LOV is generic while BioPortal is domain-specific), both show same patterns with respect to restriction types use but not to the extent they use them. *MontoloStats* reveals that both LOV and BioPortal use RDFS-based and OWL-based restriction types to a similar extent, i.e. more than 95% of ontologies use RDFS-based restrictions but only half of them use OWL-based. However, the extent of their use differs. LOV ontologies contain much more *domain* and *range* restrictions compared to BioPortal, whereas BioPortal ontologies make considerably more use of *disjointness* restrictions. Furthermore, cardinality-based restrictions seem to be preferred by LOV in their *unqualified* variant whereas BioPortal uses more *qualified* cardinalities.

We also found that different literal-value related restriction types are not used at all or to a negligible extent. This raises questions: *why is there no need to express literal-value related restrictions?*, and if there is a need *where are literal-value related restrictions currently encoded?*

Ontology reuse. MontoloStats' restriction type statistics can support ontology reuse activities concerning the assessment of relevant reuse candidates with respect to an application scenario.

MontoloStats indicates if an ontology contains e.g. a taxonomic structure (restriction type *subsumption*), or defines concepts in a machine-understandable way (using i.a. the restriction type *disjoint classes*). Such information is needed to assess the relevance of an ontology for different application scenarios, e.g. ontologies used for classification tasks ideally contain taxonomic information, but other application scenarios might rely on reasoning which likely benefits from a higher degree of axiomatization [16].

For each ontology in the *MontoloStats* dataset a dedicated website exists, listing the statistics and additional information about restriction types, i.e. definitions from their descriptions in the *Montolo* dataset. Thus, restriction type statistics can be retrieved on-demand by an ontology engineer without any additional effort with respect to the setup of a tool chain.

Ontology Engineers may perform a comparative analysis of ontology reuse candidates considering external information. *MontoloStats* and restriction type definitions in *Montolo* are available as Linked Data, and, thus, SPARQL queries can be used to retrieve and combine different data sources to semi-automatically create reusable evaluation reports.

Lessons learned and Impact. MontoloStats shows that almost half of LOV and BioPortal ontologies could be considered "*lightweight*" as they are less axiomatized. Currently, domain experts provide their knowledge and ontology engineers have to encode this knowledge in an optimal way, i.e. fulfilling all requirements while satisfying raising needs towards lightweight ontologies.

MontoloStats reveals that not all restriction types are used and those that are used are not equally used by different ontologies. We need to investigate both the roots of the observation, as well as its impact and consequences.

By comparing restriction modeling in LOV and BioPortal we found implicit modeling patterns with respect to restrictions. However, research focused on the definition of explicit methodological guidelines supporting ontology engineers in their tedious task of encoding restrictions still requires improvement. We need to better understand the restrictions and their implications compared to practical needs in an environment with changing requirements. *Are the restrictions properly modeled?*

MontoloStats reveals that not all restrictions are broadly used. However, it has not been thoroughly investigated so far how appealing ontology modeling tools are for defining restrictions. *Can the available tools support the creation of all restriction types? Are they appealing for the task at hand?*

Similarly, *MontoloStats* reveals that certain ontologies contain several restrictions and others not. However, the correlation between the number of restrictions per ontology and the ontology's reuse is not investigated so far. *Are the ontologies with restrictions and without equally (re)used? How does this influence if restrictions should be defined?* In the same context, it has not been investigated for ontologies how frequently each type of restriction is involved in knowledge graph quality issues and how this affects the evolution of the ontology. *Should we force certain restriction types found to be violated in datasets?*

Evaluation plan. Given an ontology engineering-related ontology reuse scenario, a user study could investigate to which extent *MontoloStats* improves the discovery and selection of ontologies.

Regarding *ontology-discovery*, a modified version of LOV's search interface could provide users the function to filter search results based on the existence/non-existence of restriction types or restriction type expressions. Given scenarios where more or less restrictions are desired, users can report how useful the filter-functionality based on *MontoloStats* was perceived, which restriction types they found the most useful to filter, and what information they might miss.

An *ontology-selection*-related task could similarly assess how users perceive the usefulness of *MontoloStats* when comparing ontologies. Additionally, the effectiveness of *MontoloStats* can be evaluated by comparing the amount and duration of steps to evaluate and compare ontology reuse candidates using *MontoloStats* versus manual inspection.

We plan to update the *MontoloStats* dataset regularly, but also to incorporate new restriction types and restriction type expressions into *Montolo*, identified e.g. by the community. New measures besides *occurrence* can be defined to gain a deeper understanding of restrictions use in ontologies. Last, we plan experiments to investigate the incorporation of *MontoloStats* into the LOV and BioPortal platform, to e.g. use restriction type statistics, as search filter or for results ranking.

ACKNOWLEDGMENTS

The described research activities were funded by Ghent University, imec, Flanders Innovation & Entrepreneurship (AIO), and the European Union. Ruben Verborgh is a postdoctoral fellow of the Research Foundation – Flanders. We also would like to thank the reviewers, whose comments helped to improve this work.

REFERENCES

[1] Dörthe Arndt, Ben De Meester, Anastasia Dimou, Ruben Verborgh, and Erik Mannens. 2017. Using Rule Based Reasoning for RDF Validation. In *RuleML+RR.*

[2] Sören Auer, Jan Demter, Michael Martin, and Jens Lehmann. 2012. LODStats – An Extensible Framework for High-Performance Dataset Analytics. In *Knowledge engineering and management.* Springer.

[3] Niyati Baliyan and Sandeep Kumar. 2016. Towards measurement of structural complexity for ontologies. *International Journal of Web Engineering and Technology* (2016).

[4] Antonio De Nicola and Michele Missikoff. 2016. A lightweight methodology for rapid ontology engineering. *Commun. ACM* (2016).

[5] Antonio De Nicola, Michele Missikoff, and Roberto Navigli. 2009. A software engineering approach to ontology building. *Information systems* (2009).

[6] Nicola Guarino and Pierdaniele Giaretta. 1995. Ontologies and knowledge bases: Towards a terminological clarification. In *Towards very Large Knowledge bases: Knowledge Building and Knowledge sharing.* IOS Press.

[7] Thomas Hartmann. 2016. *Validation Framework for RDF-based Constraint Languages.* Ph.D. Dissertation. Karlsruher Institut für Technologie (KIT).

[8] Dimitris Kontokostas, Patrick Westphal, Sören Auer, Sebastian Hellmann, Jens Lehmann, Roland Cornelissen, and Amrapali Zaveri. 2014. Test-driven evaluation of linked data quality. In *23rd international conference on World Wide Web.* ACM.

[9] Jose Emilio Labra Gayo, Eric Prud'hommeaux, Iovka Boneva, and Dimitris Kontokostas. 2017. *Validating RDF Data.* Morgan & Claypool Publishers LLC.

[10] Andreas Langegger and Wolfram Woss. 2009. RDFStats - An Extensible RDF Statistics Generator and Library. In *Proceedings of the 20th International Workshop on Database and Expert Systems Applications.* IEEE Computer Society.

[11] Nandana Mihindukulasooriya, Poveda-Villalón, María-Castro, Raúl, and Asunción Gómez-Pérez. 2015. Loupe-An Online Tool for Inspecting Datasets in the Linked Data Cloud. In *International Semantic Web Conference (Posters & Demos).*

[12] M Musen, N Shah, N Noy, Benjamin Dai, Michael Dorf, N Griffith, JD Buntrock, Clement Jonquet, MJ Montegut, and Daniel L Rubin. 2008. BioPortal: ontologies and data resources with the click of a mouse. In *AMIA Annu Symp Proc*, Vol. 6. 1223–1224.

[13] Mark A. Musen. 2015. The Protégé Project: A Look Back and a Look Forward. *AI Matters* (2015).

[14] James A Overton, Heiko Dietze, Shahim Essaid, David Osumi-Sutherland, and Christopher J Mungall. 2015. ROBOT: A command-line tool for ontology development.. In *ICBO.*

[15] Elena Simperl. 2009. Reusing ontologies on the Semantic Web: A feasibility study. *Data and Knowledge Engineering* (2009).

[16] Elena Simperl. 2010. Guidelines for reusing ontologies on the semantic web. *International Journal of Semantic Computing* (2010).

[17] Elena Simperl and Christoph Tempich. 2006. Ontology engineering: a reality check. In *International Conference "On the Move to Meaningful Internet Systems".* Springer.

[18] Elena Simperl, Christoph Tempich, and York Sure. 2006. ONTOCOM: a cost estimation model for ontology engineering. In *The Semantic Web - ISWC 2006.* Springer Berlin Heidelberg, Berlin, Heidelberg.

[19] Dominik Tomaszuk. 2018. Inference rules for OWL-P in N3Logic. In *Communication Papers of the 2018 Federated Conference on Computer Science and Information Systems (Annals of Computer Science and Information Systems).* PTI.

[20] Pierre-Yves Vandenbussche, Ghislain A Atemezing, María Poveda-Villalón, and Bernard Vatant. 2017. Linked Open Vocabularies (LOV): a gateway to reusable semantic vocabularies on the Web. *Semantic Web* (2017).