Observing the Impact and Adaptation to the Evolution of an Imported Ontology

Omar Qawasmeh[1], Maxime Lefrançois[2], Antoine Zimmermann[2] and Pierre Maret[1]

[1] *Univ. Lyon, CNRS, Lab. Hubert Curien, UMR 5516, F-42023 Saint-Étienne, France*

[2] *Univ. Lyon, MINES Saint-Étienne, CNRS, Lab. Hubert Curien, UMR 5516, F-42023 Saint-Étienne, France*

{*omar.alqawasmeh, pierre.maret*}*@univ-st-etienne.fr*,{*maxime.lefrancois,antoine.zimmermann*}*@emse.fr*

Keywords: Ontology, Ontology Evolution, Ontology Adaptation

Abstract: Ontology evolution is the process of maintaining an ontology up to date with respect to the changes that arise in the targeted domain or in the requirements. Inspired by this definition, we introduce two concepts related to observe the impact and the adaptation to the evolution of an imported ontology. In the first one we target the evolution of an imported ontology (if ontology O uses ontology O', and then O' evolves). The second one targets the adaptation to the evolution of the imported ontology. Based on our definition we provide a systematic categorization of the different cases that can arise during the evolution of ontologies (e.g. a term t is deleted from O', but O continues to use it). We led an experiment to identify and count the occurrences of the different cases among the ontologies referenced on two ontology portals: 1. the Linked Open Vocabulary (LOV) ontology portal which references 648 different ontologies, 88 of them evolved. We identified 74 cases that satisfy our definition, involving 28 different ontologies. 2. the BioPortal which references 770 different ontologies, 485 of them evolved. We identified 14 cases that satisfy our definition, involving 10 different ontologies. We present the observation results from this study and we show the number of different cases that occurred during the evolution. We conclude by showing that knowledge engineers could take advantage of a methodological framework based on our study for the maintenance of their ontologies.

1 INTRODUCTION

Ontologies play an important role in organizing and categorizing data in information systems and on the web. This leads to a better understanding, sharing and analyzing of knowledge in a specific domain. However, domains are subject to changes, thus arises the need to evolve ontologies in order to have an adequate representation of the targeted domain (Stojanovic et al., 2003). Ontology evolution is the process of maintaining an ontology up to date with respect to the changes that might arise in the described domain, and/or in the requirements (Zablith et al., 2015).

Re-usability is considered as a good practice while designing an ontology (Simperl, 2009). On the one hand, re-usability saves time for knowledge engineers while developing ontologies, but on the other hand it raises the problem of adapting one's ontology to the evolution of an imported ontology and thus complicates the maintenance process. In this paper we are interested in observing the impact of the evolution of an imported ontology, and the adaptation to that evolution.

For example the ontology *Schema.org core and all extension vocabularies (O)* has a version that was published in 2014-10-30 (v_1), and that uses terms from the ontology *Schema.org (O')* that was created in 2012-04-27 (v'_1). O' evolved to v'_2 in 2017-03-23, and similarly, O evolved to v_2 in 2017-05-19. The term `Bacteria` was deleted in v'_2, yet v_2 still uses it. This has an impact (O needs to be adapted), and illustrates an issue that might arise when an imported ontology evolves. We propose a frame for observing the impact and adaptation to the evolution of an imported ontology. Then we identify and try to explain occurrences of such cases in the history of two ontology portals.

The rest of the paper is organized as follows: Section 2 presents our motivating example. Section 3 presents an overview of ontology evolution. Section 4 proposes a theoretical analysis of the need for changes that could stem from the evolution of an imported ontology. Section 5 presents an exhaustive theoretical analysis of how an ontology may be adapted to such evolution. Section 6 presents our method to analyze the LOV portal and the BioPortal based on our definition. Section 7 shows the evaluation of our experiments. Section 8 discusses our results. Finally, Section 9 concludes the paper.

2 MOTIVATING EXAMPLE

Amal is a knowledge engineer who works on developing an ontology to describe the child care domain. Let's assume that she created an ontology called *Childcare*. In the version V1.1 of *Childcare*, created in May 2017, Amal used a specific term `programmOfStudy` from another ontology called *Education* created in January 2017. *Childcare* contains at least a link to a term from *Education*. In September 2017 the creators of the *Education* ontology released the version V1.2. Amal does not notice the evolution. Thus, she thinks that her ontology is still using the V1.1 version of the *Education* ontology. Several issues might arise:

1. The term `programmOfStudy` was removed from *Education*, however it is still used *Childcare*. This might have an impact over *Childcare*. As a consequence for this impact, Amal should adopt the changes and change her ontology accordingly.

2. New terms were introduced in *Education,V1.2* (e.g. `boarding school`). Amal should be made aware of these new terms in order to possibly make use of them in her ontology.

3 AN OVERVIEW ON ONTOLOGY EVOLUTION

In this section we present the life cycle of ontology evolution as defined by (Zablith et al., 2015), and some related works. The life cycle (Zablith et al., 2015) consists of five phases summarized as: 1. Detect the need for evolution by either studying the users' behavior while using ontology-based systems or by analyzing the data sources that use the ontology, 2. Suggest changes to evolve the ontology, 3. Validate the suggested changes before adopting them into the ontology, 4. Assess and study the impact of the evolution, by evaluating the impact of the changes on external artifacts that rely on the ontology (e.g. other ontologies, systems) and/or the cost of performing the changes. 5. Keep track of the implementation of the changes, in order to facilitate the management of the different versions that are created during the evolution process. In this research, we focus on both the first and fourth phases.

Authors in (Zablith et al., 2015) mentioned several research papers that tackle the problem of detecting the need of evolution. In addition, two papers (Tartir et al., 2010, Papavassiliou et al., 2009) are highly relevant for our approach that will rely on the structural changes to detect the need of the evolution (i.e. detection of addition and deletion of terms). In (Tartir et al., 2010), the authors emphasize what came from (Noy and Musen, 2002) and they mentioned that ontology evolution is caused by three reasons: 1. Changes in the described domain, 2. Changes in the conceptualization (e.g. deletion and addition), and 3. Changes in the explicit specification. In (Papavassiliou et al., 2009) a change detection algorithm is proposed and relies on a specific language they also proposed. One feature of their algorithm is to detect the need of evolution out of the changes that happen, such as renaming a class (i.e. delete and add).

Several approaches have studied the impact of ontology evolution in different scenarios. Dragoni and Ghidini (Dragoni and Ghidini, 2012) investigate how ontology evolution operations affect the effectiveness of search systems. They focused on three operations: 1. rename a concept, 2. delete a concept, and 3. move a concept. They analyzed the impact of the evolution of the ontology over a search system: they performed 75 queries over a search system at every version of the evolved ontology and they calculated the effectiveness of the system by comparing with a baseline.

Abgaz et al. (Abgaz et al., 2012) analyzed structural impact and semantic impact over ontologies. They defined a set of rules to analyze the impact by detecting unsatisfiable statements and wrong instances. They defined 10 change operations that cover the different change scenarios.

Groß et al. (Groß et al., 2012) investigated how the changes in the *Gene* ontology [1] might affect the statistical applications for the experimental and simulated data (external artifacts). CODEX tool (Hartung et al., 2012) was used to detect the changes (e.g. addition, merging, moving). They introduced their own stability measure by choosing a fixed set of genes to compute the experimental result set at different point of time with freely chosen ontology and annotation versions.

Mihindukulasooriya et al. (Mihindukulasooriya et al., 2016) introduced a study that shows how *DBpedia* (Lehmann et al., 2015), *Schema.org* (Guha et al., 2016), *PROV-O* (Lebo et al., 2013) and *FOAF* (Brickley and Miller, 2010) ontologies evolved through their life time. They counted the changes that occurred between the different versions such as, addition and deletion of classes, properties, sub-classes and sub-

[1] `http://geneontology.org/`

properties. They show that ontology evolution is more challenging when the ontology size is large. Moreover, they show the need of having tools that can help during the evolution process.

Abdel-Qader et al. (Abdel-Qader et al., 2018) analyzed the impact of the evolution of terms in 18 different ontologies referenced in LOV. Their method consisted of two phases: 1. retrieve all the ontologies that have more than one version, and 2. investigate how terms are changed and adopted in the evolving ontologies. They applied their analysis on three large-scale knowledge graphs: DyLDO [2], BTC [3] and Wikidata.[4] They found that some of the term changes in the 18 ontologies are not mapped into the three knowledge graphs. Also they concluded that there is a need for a service to keep an eye on the ontology changes. They claim that it would help the knowledge engineers and the data publishers maintaining their artifacts (other ontologies, systems or data sets).

As a conclusion for this study, we show that these different approaches studied the impact of ontology evolution by three techniques: 1. Observing the structural changes such as: addition, deletion, and moving (i.e. deletion then addition) during the life time of an ontology. 2. Measuring the impact of the evolution over the different artifacts (e.g. search systems) that might rely on the evolved ontologies. 3. Listing the changes, the frequency of each change, and the time it took to adopt a specific set of changes. In this paper, we follow the first technique, and we propose an approach that observes the evolution of terms of an imported ontology.

4 OBSERVING THE IMPACT OF THE EVOLUTION OF AN IMPORTED ONTOLOGY

As shown in the previous section, there is a need to detect when to perform changes on ontologies. This is defined in the first phase of the ontology evolution life-cycle. In this research we are interested in observing the impact of the evolution of an imported ontology, and the adaptation to that evolution. We observe the evolution of terms that have the namespace of the imported ontology. A RDF term is generally defined as: $I \cup B \cup L$[5]. In this research we take into consideration only the IRIs (I).

[2] http://km.aifb.kit.edu/projects/dyldo/data

[3] https://km.aifb.kit.edu/projects/btc-2012

[4] https://www.wikidata.org

[5] I: IRIs, B: blank nodes, and L: literals

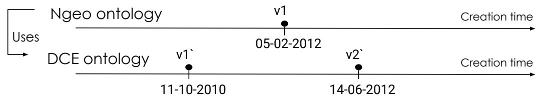

Figure 1: A time line showing the creation times of the Ngeo ontology and the DCE ontology, where Ngeo uses terms that are defined by DCE

Detecting the need for evolving the set of terms can be manifested through two behaviors:

1. There is already a problem: If an ontology O uses a term t that has the namespace of another ontology O', however it is not defined in O'.
2. A problem has occurred because of the evolution process: Let's assume that there is an ontology O that uses a term t that has the namespace of another ontology O'. O' evolved which cause the deletion of t. This evolution might cause problems for O. This raises the need to evolve O in order to reflect the new changes, which leads to solve the different problems.

We give our definition of an imported ontology evolution hereafter:

Definition 1. ***Imported ontology evolution***
Imported ontology evolution is a situation where: O is an ontology which has at least one version v_1. O' *is a different ontology which has at least two versions* v'_1 *and* v'_2. *O uses terms that have the namespace of* O'. $time(v)$ *is the creation time for a version.*

A case of imported ontology evolution is noted $\langle v_1, v'_1, v'_2 \rangle$ *and holds when the following conditions are satisfied:*
$time(v'_1) < time(v'_2) \wedge time(v'_1) < time(v_1)$

Fig 1 presents an example of one case of imported ontology evolution: the *NeoGeo Geometry* Ontology has one version (v_1: ngeo_2012-02-05) that uses the *Dublin Core Metadata Element Set* ontology (v'_1: dce_2010-10-11, v'_2: dce_2012-06-14). Table 1 reflects the different cases that may occur with respect to Definition 1. t is a term that has the namespace of O'. The circles at every line represent the set of terms (t) that exist in the the two versions of the ontology O' (i.e. v'_1 and v'_2). The columns represents the set of terms t that exist in the ontology O (i.e. v_1).

Four possible cases might happen:

Row 1. No changes over *t*

Case 1.a There is no change of t to detect, therefore there is no interest in studying this case. This case holds for all the terms t with the namespace of O', that are neither defined in O' nor used in O

Case 1.b This case holds when O uses a term t with the namespace of O', but that is not defined in O'. Some terms that have the namespace of O' are being used in v_1 without being defined before. This is a mistake, hence there is a need to evolve v_1 to reflect the latest changes.

Row 2. t **is deleted in** v_2'
The owners of O' decided to stop using a term (e.g. `programmOfStudy`) in v_2':

Case 2.a The term is not used in v_1. No problems to be reported, and v_1 was not affected by the evolution of O'.

Case 2.b During the evolution, the term t was deleted. However, it is still being used in v_1. This might introduce a problem of using terms that does not exist anymore. So v_1 should evolve to better reflect the changes of O'.

Row 3. t **exist in both** v_1' **and** v_2'
There is no changes on t:

Case 3.a The term is not used in v_1. However, it can be recommended for use in the upcoming versions of v_1.

Case 3.b No changes over the terms during the evolution.This case is not problematic.

Row 4. t **is added to** v_2'
The owners of O' introduced a new term (e.g. `boardingSchool`) in v_2':

Case 4.a The term t is not used in v_1. It can be interesting to use, thus this addition can be notified.

Case 4.b The term t is used in v_1, however it was defined later in v_2'.

Table 1: The set of cases that might happen during the evolution of O' considering a term t that has the namespace of O'

	uses O, O`	a: v1	b: v1 (t)
1	v1` v2`	Case 1.a: No changes occurred	Case 1.b: Term is used in v1 without being defined in O`
2	v1` v2` (t)	Case 2.a: No impact occurred	Case 2.b: There is a need for evolution, because the term is no longer in O`
3	v1` v2` (t)	Case 3.a: No impact occurred. Suggest to add new terms	Case 3.b: No impact occurred
4	v1` v2` (t)	Case 4.a: Suggest to add new terms	Case 4.b: Term used before it is defined

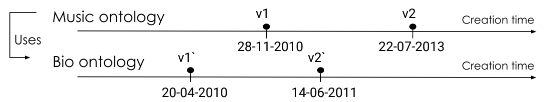

Figure 2: A time line showing the creation times of the music ontology (mo) and the bio ontology (bio), where mo uses terms that are defined by bio

5 OBSERVING THE ADAPTATION TO THE EVOLUTION OF AN IMPORTED ONTOLOGY

We have seen the impact of the evolution of an imported ontology. As a consequence of this evolution, the impacted ontology should be adapted to the changes and evolve accordingly. This creates a situation which we call ontology co-evolution.

The term "ontology co-evolution" has been already used in three research papers. Authors in (Kupfer et al., 2006, Kupfer and Eckstein, 2006) define the co-evolution as the integration between the database schemas and ontologies to design and evolve the targeted ontologies. Also (Ottens et al., 2007) defines the co-evolution as the creation of ontologies by taking advantage of natural language techniques to process some raw text. These definitions are irrelevant to the problem we investigate. We define ontology co-evolution as:

Definition 2. ***Ontology Co-Evolution***
Ontology co-evolution is a situation where: O is an ontology which has at least two versions v_1 *and* v_2. *O' is a different ontology which has at least two versions* v_1' *and* v_2'. *O uses terms that have the namespace of O'. time(v) is the creation time for a version.*

In order to have a co-evolution case between O and O' with the ontologies $\langle v_1, v_1', v_2, v_2' \rangle$, the following condition must be satisfied:
$time(v_1) < time(v_2) \wedge time(v_1') < time(v_2') \wedge time(v_1') < time(v_1) \wedge time(v_2') < time(v_2)$

As mentioned in the previous section, in this research we take into consideration only the IRIs (I). We provide an exhaustive categorization of the different cases that can arise during the adaptation to ontology evolution (i.e. co-evolution). Fig 2 presents an example of one case of ontology co-evolution: the *Music* ontology has two versions (v_1: mo_2010-11-28, v_2: mo_2013-07-22) that are respectively using the two versions of the *Bio* ontology (v_1': bio_2010-04-20, v_2': bio_2011-06-14).

During the evolution of O', terms may be introduced or deleted. We identified the occurrences of adaptation to ontology evolution of O and O'. We observe the set of terms that have the namespace of O'. Table 2 shows the different cases that may occur. The left circles represent the set of terms that exist in the first version of an ontology (i.e. v_1 and v'_1), and the right circles represent the set of terms that exist in the second version of an ontology (i.e. v_2 and v'_2). t is a term that has the namespace of O'.

In our illustrating example of Section 2, let us assume that Amal finally noticed the evolution of *Education* ontology and decided to evolve her ontology *Childcare* to V1.2 on November 2017. Based on our definition, the ontology *Childcare* is considered as O which has two versions v_1: *Childcare V1.1*, created in May 2017 and v_2: *Childcare V1.2*, created in November 2017. The ontology *Education* is considered as O' and has two versions v'_1: *Education V1.1*, created in January 2017 and v'_2: *Education V1.2*, created in September 2017. Amal is using the term `programmOfStudy` from O'. Following each line of Table 2, the following set of cases might occur during the life journey of Amal's ontology:

Row 1. No changes over the terms of v'_1, or v'_2

Case 1.a There is no change of t to detect, therefore there is no interest in studying this case.

Case 1.b Amal made a typo by using the term `programOfStudy` (i.e. program is written with one "m" instead of two) in v_1, but then she realizes that this term does not exist in O'. She fixes this mistake by not using it in v_2 anymore.

Case 1.c Amal uses t in both v_1 and v_2. This case might be explained by the fact that t is defined in a previous version (e.g. v_0) of the ontology O' (i.e. $t(v'_0) < t(v'_1)$).

Case 1.d Amal introduces a mistake by using t in v_2.

Row 2. t is deleted in v'_2
The owners of O' decided to stop using the term `programmOfStudy` in v'_2:

Case 2.a Amal does not use t that was recently deleted. Hence v_1 and v_2 were not affected.

Case 2.b Amal realizes that t was deleted, so she stops using it in v_2.

Case 2.c Amal does not realize the deletion of t, and she keeps using it in v_2.

Case 2.d Amal starts to use t in her second version (v_2), which introduces a mistake.

Row 3. t exist in both v'_1 and v'_2
None of the cases (3.a, 3.b, 3.c, and 3.d) is problematic.

Row 4. t is added to v'_2
The owners of O' introduced a new term `boardingSchool` in v'_2:

Case 4.a Amal has not noticed the addition of t, even if it might be interesting for her to introduce it.

Case 4.b Amal was already using t in (v_1), but she decided to remove it from v_2.

Case 4.c Amal was already using t in v_1, and she continues using it in v_2.

Case 4.d Amal realizes the addition of t, and she start using it in v_2.

Cases 4.b and 4.c are corner cases that are discussed further in section Section 8.

6 ANALYZING THE ADAPTATION OF THE EVOLUTION OVER ONTOLOGY PORTALS

The main aim of ontology portals is to group different ontologies in order to facilitate the process of finding and reusing them. Moreover, ontology portals are convenient to group the different versions of each ontology. These versions are time stamped in order to keep track of the creation time for each one. This facilitates our process to extract the time of each version. This section presents the two ontology portals we used and the method we applied to detect the occurrences of ontology co-evolution. Section 6.1 presents LOV portal (Vandenbussche et al., 2017), and Section 6.2 presents BioPortal (Whetzel et al., 2011). We have selected these two portals because they are the ones that reference the greatest number of vocabularies available on the Web.

6.1 Analyzing the Adaptation of Ontology Evolution over the Linked Open Vocabulary (LOV)

Linked Open Vocabulary (LOV) (Vandenbussche et al., 2017) is considered a rich repository of ontologies. LOV's main goal is to help publishers and users of linked data and vocabularies to assess, reuse, and publish different vocabularies based on their needs. LOV currently references 648 different vocabularies[6], each one being described with different properties,

[6]Last counted on June 2018

Table 2: The set of cases that might happen during the ontology co-evolution considering t that has the namespace of O'

	uses: O → O`	a	b	c	d
		v_1 v_2	v_1 t v_2	v_1 t v_2	v_1 v_2 t
1	v_1` $v_2$`	Case 1.a: No changes occurred	Case 1.b: Term is used in v_1 but doesn't exist in O`	Case 1.c: Term is used in both v_1 and v_2 but doesn't exist O`	Case 1.d: Term is used in v_2 but doesn't exist O`
2	v_1` $t$ $v_2$`	Case 2.a: Term is deleted from v_2` and not used in O	Case 2.b: Term is deleted in v_2` but still used in v_1	Case 2.c: Term is deleted in v_2` but still used in both v_1 and v_2	Case 2.d: Term is deleted in v_2` and still used in v_2
3	v_1` $t$ $v_2$`	Case 3.a: Term exists in both v_1` and $v_2$` and not used in O	Case 3.b: Term exists in both v_1` and $v_2$` and used in v_1	Case 3.c: Term exists in both v_1` and $v_2$` and used in both v_1 and v_2	Case 3.d: Term exists in both v_1` and $v_2$` and used in v_2
4	v_1` $v_2$` t	Case 4.a: Term introduced in v_2` and not used in O	Case 4.b: Term exists in v_2` and used in v_1	Case 4.c: Term exists in v_2` and used in both v_1 and v_2	Case 4.d: Term exists in v_2` and used in v_2

such as number of incoming links (i.e. how many ontologies are using ontology O), number of outgoing links (i.e. how many ontologies are used by ontology O), number of different versions, and datasets that are using ontology O.

Out of the 648 ontologies of LOV, there are 88 ontologies that have evolved with a total number of 344 versions. The number of different versions that is associated with each ontology varies. For example the *FOAF*[7] ontology has 10 available versions to download from the LOV knowledge base. Fig 3 shows the relation between the total number of versions and the number of ontologies that have the specific number of versions.

However, not all the ontologies that evolved are connected (i.e. using terms from each other). In order to retrieve the set of ontologies that satisfy the conditions in Definition 2, we first issued a SPARQL query (Listing 1.1) over the LOV RDF dump in order to retrieve all ontologies that have at least two different versions and at least 1 incoming link. The result is 46 different ontologies and a total of 205 different versions.

Second, we used Apache-Jena in order to get all the different ontologies that have more than one outgoing links. This decreased the list of versions to 198. Third, we extracted all the creation times for the different ontologies versions, and we filtered them based on the selection criteria of Definition 2. As a result

[7]Available at: `https://lov.linkeddata.es/dataset/lov/vocabs/foaf`

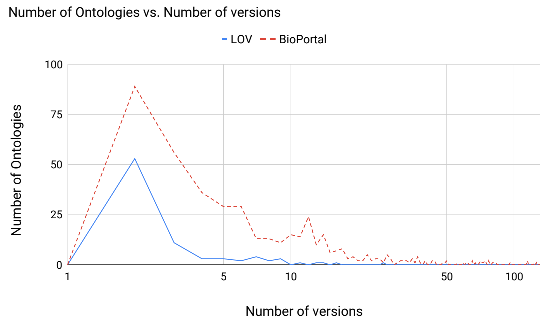

Figure 3: The relation between the total number of versions and the number of ontologies that have the specific number of versions for the ontologies that are referenced in LOV and BioPortal

we identified 74 cases of ontology co-evolution, involving 28 different ontologies.

6.2 Analyzing the Adaptation of Ontology Evolution over BioPortal

BioPortal (Whetzel et al., 2011) is an open repository for biomedical ontologies. BioPortal's main goal is to make biomedical knowledge and data available on the internet using ontologies. This is useful for boosting biomedical science and clinical care domains. BioPortal currently references 770 different ontologies,[8] each one being described with different properties,

[8]Last counted on January 2019

```
PREFIX voaf:<http://purl.org/vocommons/voaf#>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX dcat: <http://www.w3.org/ns/dcat#>

SELECT DISTINCT * WHERE {
 GRAPH <http://lov.okfn.org/dataset/lov>{

 ?vocab a voaf:Vocabulary ;
  dcterms:title ?title ;
  dcterms:modified ?modified ;
  voaf:reusedByDatasets ?dataset ;
  voaf:reusedByVocabularies ?import ;
  dcat:distribution ?distrib .

 ?distrib dcterms:issued ?issued .

 FILTER ( !isBlank(?distrib) )
 FILTER ( ?import >0 )
 BIND (STR(?issued) AS ?date)
 }
} ORDER BY DESC(?dataset)
```

Listing 1: This query returns all ontologies which have at least 2 versions and at least 1 incoming link

such as the number of different versions, along with general metrics (e.g. number of classes, properties and instances).

Out of the 770 ontologies of BioPortal, 485 ontologies have evolved with a total number of 15025 versions. The number of different versions that is associated with each ontology varies. For example the *HIV* [9] ontology has 12 available versions. Fig 3 shows the relation between the total number of versions and the number of ontologies that are associated with each ontology. As explained earlier, only some of the ontologies that evolved satisfy our configuration of ontology co-evolution.

In order to retrieve the set of ontologies that satisfy the conditions in Definition 2, we firstly used the BioPortal API [10] to retrieve all ontologies that have at least two versions. As a result, we got 485 ontologies that have 15025 different versions. Secondly, we used Apache-Jena in order to get all different ontologies that have more than one outgoing link. And thirdly, we extracted all the creation times for the different ontology versions, and we filtered them based on the selection criteria of Definition 2. We identified 14 cases of ontology co-evolution, involving 10 different ontologies.

[9] Available at: `http://data.bioontology.org/ontologies/HIV/submissions`

[10] `http://data.bioontology.org/documentation`

7 IDENTIFICATION OF THE OCCURRENCES OF ADAPTATION TO ONTOLOGY EVOLUTION

In this section, we present the results of an experiment[11] to detect ontology co-evolution using the cases that are defined in Section 5. In Section 8 we discuss these results in more details.

We retrieved from LOV a set of 28 ontologies with 74 co-evolution instances. As for BioPortal we retrieved a set of 10 ontologies with 14 co-evolution instances. [12]

We extracted the set of terms for each version, and the name spaces for the used ontologies (O''s versions), and we used them to compute the number of occurrences of the different cases.

Table 3 shows the number of occurrences for each co-evolution case for LOV (first value in each cell) and BioPortal (second value).

Table 3: The number of occurrences for each co-evolution case for LOV (first value) and BioPortal (second value) with respect to namespace of (O').

	uses O / O′	a (v1 v2)	b (v1 t v2)	c (v1 t v2)	d (v1 v2 t)
1	v1′ v2′		0 130	3 929	3 115
2	v1′ t v2′	23 27	0 0	0 3	0 0
3	v1′ t v2′	16875 9135	10 0	270 2058	23 0
4	v1′ v2′ t	2420 1560	0 115	0 908	0 0

Now we discuss further different interesting cases. In the following subset of cases, a version of O uses a term t with the namespace of O', however t is not defined in the two versions of O':

Case 1.b (i.e. a term t is used in v_1, however it does not exist in v'_1 and v'_2): This co-evolution case occurred 130 times in BioPortal but never in LOV. An example is the co-evolution process of the *Schema.org core and all extension vocabularies* (v_1: created in 2014-10-30 and v_2: created in 2017-05-19), with *Schema.org* ontology (v'_1: created in 2012-04-27 and v'_2: created in 2017-03-23). The terms `Bacteria`,

[11] Can be found at: `https://github.com/OmarAlqawasmeh/coEvolutionTermsExtraction` (Full results can be found inside *resources* folder)

[12] The co-evolution cases of LOV and BioPortal are inside the *resources* folder at `https://github.com/OmarAlqawasmeh/coEvolutionTermsExtraction`

FDAcategoryC and Diagnostic are used in v_1, however they do not exist in v'_1 and v'_2.

Case 1.c (i.e. a term t is used in both v_1 and v_2, however it does not exist in v'_1 and v'_2): This case occurred 3 times in LOV and 929 times in BioPortal. An example is the co-evolution process of the *Statistical Core Vocabulary* (v_1: created in 2011-08-05 and v_2: created in 2012-08-09), with *DCMI Metadata Terms* (v'_1: created in 2010-10-11 and v'_2: created in 2012-06-14). The terms dc:status and dc:partOf are used in v_1 and v_2, however they do not exist in v'_1 and v'_2.

Case 1.d (i.e. a term t is used in v_2, however it does not exist in v'_1 and v'_2): This case occurred 3 times in LOV and 115 times in BioPortal. An example is the co-evolution process of the *Europeana Data Model vocabulary* (v_1: created in 2012-01-23 and v_2: created in 2013-05-20), with *Dublin Core Metadata Element Set* (v'_1: created in 2010-10-11 and v'_2: created in 2012-06-14). The terms dc:issued and dc:modified are used in v_2 however they do not exist in v'_1 and v'_2.

In the following subset of cases a term t is defined in v'_1 and is deleted in v'_2:

Case 2.a (i.e. a term t is deleted in v'_2, and it is not used in any of O's versions): This case occurred 23 times in LOV and 27 times in BioPortal. This is a normal case, and no problem occurred during the co-evolution.

Case 2.b (i.e. a term t is deleted in v'_2, and then deleted in v_2): This case has no occurrences in LOV nor in BioPortal. We are not discussing it further.

Case 2.c (i.e. a term t is deleted in v'_2, however it is used in v_1 and still in v_2): This case occurred 3 times in BioPortal. It shows a problem of using terms that do not exist anymore in O'. For example in the co-evolution process of the *Schema.org core and all extension vocabularies* (v_1: created in 2014-10-30 and v_2: created in 2017-05-19), with *Schema.org* ontology (v'_1: created in 2012-04-27 and v'_2: created in 2017-03-23). The terms MedicalClinic, Optician and VeterinaryCare are used in both v_1 and v_2, however they do not exist in the latest version of O' (these different terms were deleted from v_2 of *Schema.org*).

Case 2.d (i.e. a term t is deleted in v'_2, however it is added in v_2): This case has no occurrences in LOV nor in BioPortal. We are not discussing it further.

Cases 3.a, 3.b, 3.c, and 3.d are not problematic cases, so they are not investigated further.

In the following subset of cases a term t is introduced in v'_1:

Case 4.a (i.e. a term t is added in v'_2, and it was not used in v_1 or v_2): There were 2420 terms added to v'_2 in the ontologies that are referenced in LOV, and 1560 terms were added to v'_2 in the ontologies that are referenced in BioPortal. These different terms are not used in v_1 or v_2. An example is the co-evolution process of the *Semanticscience Integrated Ontology (SIO)* (v_1: created in 2015-06-24 and v_2: created in 2015-09-02), with *The Citation Typing Ontology (CITO)* (v'_1: created in 2010-03-26 and v'_2: created in 2015-07-03). The term isDocumentedBy from O' could be useful to use by O, thus the owners of O can be notified and recommended to use it.

Case 4.b (i.e. a term t is added in v'_2, and it was already used in v_1): This case occurred 115 times in BioPortal, and it has no occurrence in LOV. An example is the co-evolution process of the *Schema.org core and all extension vocabularies* (v_1: published in 2014-10-30 and v_2: published in 2017-05-19), with *Schema.org* ontology (v'_1: created in 2012-04-27 and v'_2: created in 2017-03-23). The terms SoundtrackAlbum, Hardcover and SingleRelease are used in v_1, however they were introduced later in v'_2.

Case 4.c (i.e. a term t is added in v'_2, and it was already used in both of O's versions): This case occurred 951 times in BioPortal, and it has no occurrence in LOV. An example is the co-evolution process of the *Semanticscience Integrated Ontology (SIO)* (v_1: created in 2015-06-24 and v_2: created in 2015-09-02), with *The Citation Typing Ontology (CITO)* (v'_1: created in 2010-03-26 and v'_2: created in 2015-07-03). The term citesAsAuthority is used in both v_1 and v_2, however it was introduced in v'_2.

Case 4.d (i.e. a term t is added in v'_2, and v_2 starts to use it): This case has no occurrences in LOV or BioPortal, so we are not investigating them.

8 DISCUSSION

Our approach relies on the described situation in Definition 2. However we can stress some comparison with (Abdel-Qader et al., 2018):

1. Both analyses observe the additions and deletions of terms, however in (Abdel-Qader et al., 2018) they observe how the terms are changed and adopted in the evolving ontologies, where we observe the changes when two ontologies are connected to each other.

2. 13 ontologies with 37 versions that are referenced in LOV were used in the experiments of (Abdel-Qader et al., 2018), where we retrieved a total of 38 ontologies referenced in LOV and BioPortal and we observed 88 evolution cases.

After analyzing the results, we confirm the observation of (Groß et al., 2012, Kirsten et al., 2009) showing that in general the addition of terms occurs more frequently than the deletion of terms during the evolution process. Table 4 shows the number of added terms comparing to the number of deleted terms for the set of ontologies that are referenced in LOV and BioPortal and satisfy our definition of co-evolution.

Table 4: Number of added terms comparing to number of deleted terms in both LOV and BioPortal

Portal	LOV		BioPortal	
	Added	Deleted	Added	Deleted
v_2	26	10	115	245
v'_2	2420	23	2583	30

Our results can be conveniently presented in three categories that clearly appear in our experiment: 1. Assessment of Good Practices 2. Detection of Wrong Practices 3. Uncertain cases. In the next subsections we discuss these different categories.

8.1 Assessment of Good Practices

Case 1.b in BioPortal shows a good practice from the owners of O. They noticed that the term t is not defined in both v'_1 and v'_2, so they decided to delete it from v_2.

Case 2.a in LOV and Bioportal show a good practice from the owners of O'. They noticed that the term t is not used in both v_1 and v_2 so they decided to delete it from v'_2.

Cases 2.b and 2.d have no occurrences in all of the ontologies that are referenced in both LOV and BioPortal. This is the normal case of ontology evolution, and it is one indicator of the quality of the co-evolution. For instance, case 2.b indicates that the set of ontologies stops using the terms after they have been deleted in O', and case 2.d indicates that there were no mistake of using the set of deleted terms in the newest version of O.

8.2 Detection of Wrong Practices

Cases (1.c and 1.d) from LOV and cases (1.c and 1.d) from BioPortal, demonstrate the problem of using terms that do not exist in v'_1 and v'_2. A possible explanation is that these terms were used from a previous version of O'. Let's assume that this previous version is v'_0, then these cases can happen only if the publishing time of $t(v'_0)$ is before the publishing time of $t(v'_1)$. In these cases, the owners of O, should be notified of the changes, and they should be suggested to delete the terms that do not exist anymore.

Case 2.c from BioPortal shows that some terms are still used in both of O's versions after being deleted from O'. In order to prevent such kind of problems the owners should be notified about these cases.

Case 4.b from BioPortal shows that some terms have been already used in v_1, however they were added later in v'_2. The v_1 of *Schema.org core and all extension vocabularies* uses terms that were later defined by v'_2 of *Schema.org* ontology. The *Schema.org core and all extension vocabularies* is an extension of *Schema.org*, however it has its own namespace. Each reviewed extension for schema.org has its own chunk of schema.org namespace (e.g. if extension name is x, the namespace of this extension is x1.schema.org). [13] We retrieved all terms that have the namespace of *Schema.org*.[14] Other terms with different namespaces were discarded. [15] This reflects a bad practice in a way of using terms that have not been defined in the second version. These terms could be harbinger to add in the next versions.

Case 4.c from BioPortal shows that some terms are used in v_1 and v_2, however they were firstly introduced in v'_2. Both of v_1 and v_2 of *Semanticscience Integrated Ontology (SIO)* use terms that were later defined by v'_2 of *The Citation Typing Ontology (CITO)*. The term `citesAsAuthority` was firstly defined in v'_2 *The Citation Typing Ontology (CITO)*, however there is an object property that has the same name in v'_1. One explanation for this kind of errors is that the knowledge engineers might introduce a typos during the development process of the ontology. In these cases, the owners of O, should be notified that the term they use is not a term. They should look at it carefully and possibly delete it.

8.3 Uncertain cases

In cases (3.a, 3.b, 3.c and 3.d) from LOV and Bioportal, there was no change of terms in the two versions of O'. This indicates that the co-evolution process has no problem to report. Some terms are shared between v'_1 and v'_2 so there was no addition or deletion over them.

Cases 4.a and 4.d in both LOV and BioPortal show the number of terms that were added during the evolution of O'. These terms were not used in any of O's

[13]More details about the extensions managing of schema.org can be found at: `https://schema.org/docs/extension.html`

[14]namespace of *Schema.org* is http://schema.org/

[15]Some examples of discarded namespaces: https://health-lifesci.schema.org/, https://pending.schema.org/, https://meta.schema.org/

versions. These cases can be explained in two ways:

1. The owners of O did not notice the addition of these terms, however they might be interested in using some of these new terms. This might introduce a problem, thus further content analysis should be introduced to possibly recommend changes to the owners.

2. The owners of O noticed the addition of these terms and they decided not to add them.

9 Conclusion and Future work

As shown from the literature review, there is a need to formalize a conceptual frame for assessing the impact of ontology evolution and for tackling the different issues that arise during the evolution of ontologies. In this paper we presented a situation of ontology evolution which considers the evolution of an ontology O that imports another one O' (i.e. O uses terms that have the namespace of O'). We provide an exhaustive categorization of the adaptation to ontology evolution for this situation. We observe these cases over two ontology portals:

1. The Linked Open Vocabulary (LOV) ontology portal which references 648 different ontologies, 88 of them evolved. We identified 74 cases of ontology co-evolution, involving 28 different ontologies (Section 6.1).

2. The BioPortal which references 770 different ontologies, 485 of them evolved. We identified 14 cases of ontology co-evolution, involving 10 different ontologies (Section 6.2).

The usage of ontologies is increasing, so there is the need of managing them, especially in the evolution process. The main aim of this research is to introduce fundamentals for a methodological framework for ontology management during the ontology evolution. Having such kind of frameworks will effectively help to automate the process of managing ontologies during their evolution cycle which can lead to save time and effort.

We emphasize the need of having a service that can automatically observe and notify the ontologies' owners during the evolution process. Having such tool can help to keep track of the different ontologies during the co-evolution and help to facilitate the process of ontology evolution.

REFERENCES

Abdel-Qader, M., Scherp, A., and Vagliano, I. (2018). Analyzing the evolution of vocabulary terms and their impact on the LOD cloud. In *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings*, pages 1–16.

Abgaz, Y. M., Javed, M., and Pahl, C. (2012). Analyzing impacts of change operations in evolving ontologies. In *Proceedings of the 2nd Joint Workshop on Knowledge Evolution and Ontology Dynamics, USA, November 12, 2012*.

Brickley, D. and Miller, L. (2010). FOAF vocabulary specification 0.91.

Dragoni, M. and Ghidini, C. (2012). Evaluating the impact of ontology evolution patterns on the effectiveness of resources retrieval. In *Proceedings of the 2nd Joint Workshop on Knowledge Evolution and Ontology Dynamics, Boston, MA, USA, November 12, 2012*.

Groß, A., Hartung, M., Prüfer, K., Kelso, J., and Rahm, E. (2012). Impact of ontology evolution on functional analyses. *Bioinformatics*, 28(20):2671–2677.

Guha, R. V., Brickley, D., and Macbeth, S. (2016). Schema.org: evolution of structured data on the web. *Communications of the ACM*, 59(2):44–51.

Hartung, M., Groß, A., and Rahm, E. (2012). CODEX: exploration of semantic changes between ontology versions. *Bioinformatics*.

Kirsten, T., Hartung, M., Groß, A., and Rahm, E. (2009). Efficient management of biomedical ontology versions. In *On the Move to Meaningful Internet Systems: OTM 2009 Workshops, Confederated International Conferences, Portugal*, pages 574–583.

Kupfer, A. and Eckstein, S. (2006). Coevolution of database schemas and associated ontologies in biological context. In *22nd British National Conference on Databases*.

Kupfer, A., Eckstein, S., Neumann, K., and Mathiak, B. (2006). A coevolution approach for database schemas and related ontologies. In *19th IEEE International Symposium on Computer-Based Medical Systems (CBMS 2006), Salt Lake City, Utah, USA*, pages 605–610.

Lebo, T., Sahoo, S., and McGuinness, D. (2013). PROV-O: The PROV ontology. *W3C recommendation*, 30.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., and Bizer,

C. (2015). Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Mihindukulasooriya, N., Poveda-Villalón, M., García-Castro, R., and Gómez-Pérez, A. (2016). Collaborative ontology evolution and data quality - an empirical analysis. In *13th International Workshop, OWLED, and 5th International Workshop, Bologna, Italy*, pages 95–114.

Noy, N. F. and Musen, M. A. (2002). PROMPTDIFF: A fixed-point algorithm for comparing ontology versions. In *The Eighteenth National Conference on AI and Fourteenth Conference on Innovative Applications of AI, Canada.*

Ottens, K., Aussenac-Gilles, N., Gleizes, M. P., and Camps, V. (2007). Dynamic ontology co-evolution from texts: Principles and case study. In *Proceedings of the First International Workshop on Emergent Semantics and Ontology Evolution, ESOE 2007, co-located with ISWC 2007 + ASWC 2007, Busan, Korea*, pages 70–83.

Papavassiliou, V., Flouris, G., Fundulaki, I., Kotzinos, D., and Christophides, V. (2009). On detecting high-level changes in RDF/S kbs. In *ISWC 2009, 8th International Semantic Web Conference,USA.*, pages 473–488.

Simperl, E. P. B. (2009). Reusing ontologies on the semantic web: A feasibility study. *Data Knowl. Eng.*, 68(10):905–925.

Stojanovic, L., Maedche, A., Stojanovic, N., and Studer, R. (2003). Ontology evolution as reconfiguration-design problem solving. In *Proceedings of the 2nd International Conference on Knowledge Capture (K-CAP 2003), October 23-25, 2003, Sanibel Island, FL, USA*.

Tartir, S., Arpinar, I. B., and Sheth, A. P. (2010). Ontological evaluation and validation. In *Theory and applications of ontology: Computer applications*, pages 115–130. Springer.

Vandenbussche, P., Atemezing, G., Poveda-Villalón, M., and Vatant, B. (2017). Linked open vocabularies (LOV): A gateway to reusable semantic vocabularies on the web. *Semantic Web*.

Whetzel, P. L., Noy, N. F., Shah, N., Alexander, P. R., Dorf, M., Fergerson, R. W., Storey, M. D., Smith, B., Chute, C. G., and Musen, M. A. (2011). Bioportal: Ontologies and integrated data resources at the click of a mouse. In *Proceedings of the 2nd International Conf. on Biomedical Ontology, Buffalo, NY, USA*.

Zablith, F., Antoniou, G., d'Aquin, M., Flouris, G., Kondylakis, H., Motta, E., Plexousakis, D., and Sabou, M. (2015). Ontology evolution: a process-centric survey. *Knowledge Eng. Review*, 30(1):45–75.