Supplemental material

In this supplemental material, we give more details about algorithms and experiments, implemented systems as well as a note on removing versus debugging and a derivation of the Hasse diagrams.

A. Note about removing versus debugging
B. Normalization algorithm
C. Algorithms for combination strategies
D. Experiment - data
E. Experiment - results
F. Implemented systems
G: Derivation of Hasse diagrams

A. Note about removing versus debugging

We note that in this paper we deal with removing axioms with the assumption that when they are removed, they cannot be derived from the TBox representing the ontology anymore. This is not the full debugging problem, for which the combination with weakening and completing is left for future work. Removing can be seen as a simple kind of debugging, or as the second step of the debugging process. As an example, assume a TBox with axioms $A \sqsubseteq B$, $B \sqsubseteq C$, and $A \sqsubseteq C$. Assume that $A \sqsubseteq B$ and $A \sqsubseteq C$ are wrong axioms. In full debugging it would be possible to set $W = \{A \sqsubseteq C\}$. The system would then compute that $A \sqsubseteq C$ as well as one of $A \sqsubseteq B$ and $B \sqsubseteq C$ need to be removed. A domain expert may then choose to remove $A \sqsubseteq B$ and $A \sqsubseteq C$. In our problem statement it is not possible that $W = \{A \sqsubseteq C\}$ as, when removing $A \sqsubseteq C$, it still can be derived from the remaining axioms. We thus assume that a first debugging step has been performed, e.g., using traditional methods, and then start with $W = \{A \sqsubseteq C, A \sqsubseteq B\}$. Combining full debugging, weakening and completing will add additional complexity to the already complex problem we describe in this paper.

B. Normalization algorithm

Algorithm 1 rewrites an axiom into one of the allowed forms.

Algorithm 1 Normalize($sb \sqsubseteq sp$)

Input: Axiom $sb \sqsubseteq sp$
Output: A set of axioms in normalized form

if $sp \in N_c$ **then**
  return $\{ sb \sqsubseteq sp \}$
else if sp is of the form $P \sqcap Q$ **then**
  return $\{ sb \sqsubseteq P,\ sb \sqsubseteq Q \}$
else if sp is of the form $\exists r.P$ **then**
  if $sb \in N_c$ **then**
    return $\{ sb \sqsubseteq sp \}$
  else if sb is of the form $\exists r.Q$ **then**
    Introduce new concept Z
    return $\{ \exists r.Q \sqsubseteq Z,\ Z \sqsubseteq \exists r.Q,\ Z \sqsubseteq sp\}$
  else if sb is of the form $\exists s.Q$ **then**
    Introduce new concept Z
    return $\{ \exists s.Q \sqsubseteq Z,\ Z \sqsubseteq \exists s.Q,\ Z \sqsubseteq sp\}$
  else if sb is of the form $Q \sqcap R$ **then**
    Introduce new concept Z
    return $\{ Q \sqcap R \sqsubseteq Z,\ Z \sqsubseteq Q,\ Z \sqsubseteq R,\ Z \sqsubseteq sp \}$
  end if
end if

C. Algorithms for combination strategies

In this part, we give more details about the different algorithms used in the experiments. We show all our algorithms for combining different removing, weakening and completing strategies including the ones that were presented in the paper earlier. A brief description of each algorithm is shown in Table 1.

Algorithm C1 Weaken one at a time, add weakened axiom sets and remove all wrong at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
end for
$\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}$,$\bigcup_{\alpha \sqsubseteq \beta} w_{\alpha \sqsubseteq \beta}$)
return Remove-axioms($\mathcal{T}_r$,W)

Algorithm C2 Remove/weaken/add weakened axiom sets one at a time

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

$\mathcal{T}_r \leftarrow \mathcal{T}$
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}_r$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
  $\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}_r$,$w_{\alpha \sqsubseteq \beta}$)
end for
return $\mathcal{T}_r$

Algorithm C3 Remove all wrong, weaken all and add weakened axiom sets at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

$\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, W)
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
end for
return Add-axioms($\mathcal{T}_r$,$\bigcup_{\alpha \sqsubseteq \beta} w_{\alpha \sqsubseteq \beta}$)

Algorithm C4 Remove all wrong, weaken/add weakened axiom sets one at a time

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

$\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, W)
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
  $\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}_r$,$w_{\alpha \sqsubseteq \beta}$)
end for
return $\mathcal{T}_r$

Algorithm C5 Weaken one at a time, complete one at a time, add completed axiom set and remove all wrong at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
end for
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $c_{\alpha \sqsubseteq \beta} \leftarrow \emptyset$
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}, Or$)
    $c_{\alpha \sqsubseteq \beta} \leftarrow c_{\alpha \sqsubseteq \beta} \cup c_{sb \sqsubseteq sp}$
  end for
end for
$\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}$,$\bigcup_{\alpha \sqsubseteq \beta} c_{\alpha \sqsubseteq \beta}$)
return Remove-axioms($\mathcal{T}_r$,W)

Algorithm C6 Weaken/complete/add completed axiom sets one at a time, remove all wrong at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
  $c_{\alpha \sqsubseteq \beta} \leftarrow \emptyset$
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}_r, Or$)
    $\mathcal{T} \leftarrow$ Add-axioms($\mathcal{T}$, $c_{sb \sqsubseteq sp}$)
  end for
end for
return Remove-axioms($\mathcal{T}$,W)

Algorithm C7 Remove/weaken/complete/add completed axiom sets one at a time

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

$\mathcal{T}_r \leftarrow \mathcal{T}$
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}_r$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
  $c_{\alpha \sqsubseteq \beta} \leftarrow \emptyset$
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}_r, Or$)
    $\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}_r$, $c_{sb \sqsubseteq sp}$)
  end for
end for
return $\mathcal{T}_r$

Algorithm C8 Weaken/complete one at a time, add completed axiom sets and remove all wrong axioms at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
  $c_{\alpha \sqsubseteq \beta} \leftarrow \emptyset$
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}, Or$)
    $c_{\alpha \sqsubseteq \beta} \leftarrow c_{\alpha \sqsubseteq \beta} \cup c_{sb \sqsubseteq sp}$
  end for
end for
$\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}$,$\bigcup_{\alpha \sqsubseteq \beta} c_{\alpha \sqsubseteq \beta}$)
return Remove-axioms($\mathcal{T}_r$,W)

Algorithm C9 Weaken one at a time, remove all wrong, complete one at a time, then add completed axiom sets at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
end for
$\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}_r$,W)
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $c_{\alpha \sqsubseteq \beta} \leftarrow \emptyset$
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}_r, Or$)
    $c_{\alpha \sqsubseteq \beta} \leftarrow c_{\alpha \sqsubseteq \beta} \cup c_{sb \sqsubseteq sp}$
  end for
end for
$\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}_r$,$\bigcup_{\alpha \sqsubseteq \beta} c_{\alpha \sqsubseteq \beta}$)
return $\mathcal{T}_r$

Algorithm C10 Weaken one at a time, remove all wrong, complete/add completed axiom sets one at a time

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
end for
$\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, W)
for each $\alpha \sqsubseteq \beta \in W$ **do**
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}_r, Or$)
    $\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}_r$,$c_{sb \sqsubseteq sp}$)
  end for
end for
return $\mathcal{T}_r$

Algorithm C11 Remove/Weaken one at a time, add the wrong axiom and then complete/add completed axiom sets one at a time, remove all wrong at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
end for
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, $\{\alpha \sqsubseteq \beta\}$)
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}_r, Or$)
    $\mathcal{T} \leftarrow$ Add-axioms($\mathcal{T}$,$c_{sb \sqsubseteq sp}$)
  end for
end for
return Remove-axioms($\mathcal{T}$,W)

Algorithm C12 Remove all wrong, weaken all, complete all, add completed axiom sets at end

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

$\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, W)
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
end for
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $c_{\alpha \sqsubseteq \beta} \leftarrow \emptyset$
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}_r, Or$)
    $c_{\alpha \sqsubseteq \beta} \leftarrow c_{\alpha \sqsubseteq \beta} \cup c_{sb \sqsubseteq sp}$
  end for
end for
return Add-axioms($\mathcal{T}_r$,$\bigcup_{\alpha \sqsubseteq \beta} c_{\alpha \sqsubseteq \beta}$)

Algorithm C13 Remove all wrong, weaken/complete/add completed axiom sets one at a time

Input: TBox $\mathcal{T}$, Oracle Or, set of unwanted axioms W
Output: A repaired TBox

$\mathcal{T}_r \leftarrow$ Remove-axioms($\mathcal{T}$, W)
for each $\alpha \sqsubseteq \beta \in W$ **do**
  $w_{\alpha \sqsubseteq \beta} \leftarrow$ weakened-axiom-set($\alpha \sqsubseteq \beta$, $\mathcal{T}_r, Or$)
  for each $sb \sqsubseteq sp \in w_{\alpha \sqsubseteq \beta}$ **do**
    $c_{sb \sqsubseteq sp} \leftarrow$ completed-axiom-set($sb \sqsubseteq sp$, $\mathcal{T}_r, Or$)
    $\mathcal{T}_r \leftarrow$ Add-axioms($\mathcal{T}_r$,$c_{sb \sqsubseteq sp}$)
  end for
end for
return $\mathcal{T}_r$

Table 1. Algorithms.

Algorithm	Description
C1	Weaken one at a time, add weakened axiom sets and remove all wrong at end
C2	Remove/weaken/add weakened axiom sets one at a time
C3	Remove all wrong, weaken one at a time, add weakened axiom sets at end
C4	Remove all wrong, weaken/add weakened axiom sets one at a time
C5	Weaken one at a time, complete one at a time, add completed axiom sets and remove all wrong at end
C6	Weaken/complete/add completed axiom sets one at a time, remove all wrong at end
C7	Remove/weaken/complete/add completed axiom sets one at a time
C8	Weaken/complete one at a time, add completed axiom sets and remove all wrong at end
C9	Weaken one at a time, remove all wrong, complete one at a time, then add completed axiom sets at end
C10	Weaken one at a time, remove all wrong, complete/add completed axiom sets one at a time
C11	Weaken one at a time, complete/add completed axiom sets one at a time, remove all wrong at end
C12	Remove all wrong, weaken all, complete all, add completed axiom sets at end
C13	Remove all wrong, weaken/complete/add completed axiom sets one at a time

D. Experiment - data

The Mini-GALEN ontology

We repeat here the Min-GALEN ontology that is used in the main paper (Figure 1) and give a visualization (Figure 2).

N_C = {GPr (GranulomaProcess), NPr (NonNormalProcess),
PPh (PathologicalPhenomenon), F(Fracture), E (Endocarditis),
IPr (InflammationProcess), PPr (PathologicalProcess),
C (Carditis), CVD (CardioVascularDisease)};
N_R = { hAPr (hasAssociatedProcess) }
$\mathcal{T}$ = { CVD $\sqsubseteq$ PPh, F $\sqsubseteq$ PPh, $\exists$hAPr.PPr $\sqsubseteq$ PPh, E $\sqsubseteq$ C,
E $\sqsubseteq$ $\exists$hAPr.IPr, GPr $\sqsubseteq$ NPr, PPr $\sqsubseteq$ IPr, IPr $\sqsubseteq$ GPr, E $\sqsubseteq$ PPr };
W = { E $\sqsubseteq$ PPr, PPr $\sqsubseteq$ IPr, IPr $\sqsubseteq$ GPr }
Or returns *true* for:
GPr $\sqsubseteq$ IPr, GPr $\sqsubseteq$ PPr, GPr $\sqsubseteq$ NPr, IPr $\sqsubseteq$ PPr, IPr $\sqsubseteq$ NPr,
PPr $\sqsubseteq$ NPr, CVD $\sqsubseteq$ PPh, F $\sqsubseteq$ PPh, E $\sqsubseteq$ PPh, E $\sqsubseteq$ C,
E $\sqsubseteq$ CVD, C $\sqsubseteq$ PPh, C $\sqsubseteq$ CVD, $\exists$hAPr.PPr $\sqsubseteq$ PPh,
$\exists$hAPr.IPr $\sqsubseteq$ PPh, E $\sqsubseteq$ $\exists$hAPr.IPr, E $\sqsubseteq$ $\exists$hAPr.PPh.
Note that for an oracle that does not make mistakes,
if Or(P $\sqsubseteq$ Q) = true, then also Or($\exists$r.P $\sqsubseteq$ $\exists$r.Q)=true and
Or(P $\sqcap$ O $\sqsubseteq$ Q)=true.
For other axioms P $\sqsubseteq$ Q with P, Q $\in$ N_C, Or(P $\sqsubseteq$ Q) = false.

Fig. 1. Mini-GALEN; same as in main paper. (Visualized in Figure 2.)

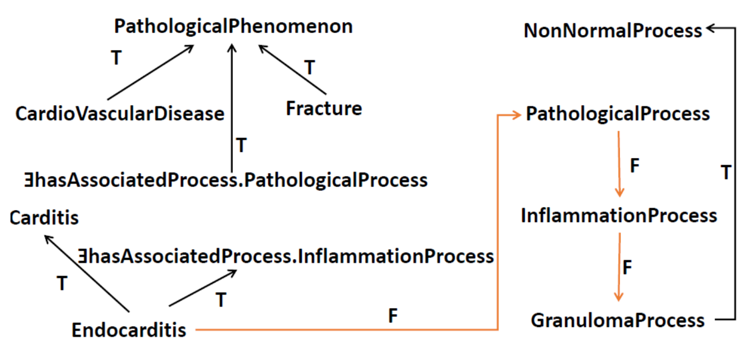

Fig. 2. Visualization of the Mini-GALEN ontology in Figure 1. The axioms in the TBox are represented with black arrows except for the wrong axioms which are represented in red. The oracle's knowledge about the axioms in the ontology is marked with T (true) or F (false) at the arrows.

Characteristics of the 6 ontologies in the experiments

In order to compare the use of the different combinations of strategies, we run experiments on several ontologies: Mini-GALEN, PACO, NCI, EKAW, OFSMR and Pizza ontology. Mini-GALEN is an example inspired by the GALEN (from `https://bioportal.bioontology.org/`) ontology; PACO, NCI, OFSMR are available at `https://bioportal.bioontology.org/`, EKAW from the conference track of `http://oaei.ontologymatching.org/` and the Pizza ontology is available at `https://github.com/owlcs/pizza-ontology`. We made the versions of the ontologies that we used available at `https://figshare.com/s/f3b9472a7e5dd69237dc`. We have used the parts of these ontologies that are expressible in $\mathcal{EL}$ in the sense that we removed the parts of axioms that used constructors not in $\mathcal{EL}$.

An overview of the numbers of concepts, roles and axioms in these ontologies is given in Table 2.

Table 2. Ontologies

	Mini-GALEN	Pizza	EKAW	OFSMR	PACO	NCI
Concepts	9	74	100	159	224	3304
Roles	1	33	8	2	23	1
Axioms	20	341	801	1517	1153	30364

E. Experiment - results

In this part, we give the full results of the comparative experiments run in different ontologies.

Table 3 lists the wrong axioms we introduced in each test ontology for experiments. These wrong axioms were generated by replacing existing axioms with axioms where their left/right-hand side concepts were changed. Figure 2 visualizes the structure of the Mini-Galen ontology in Figure 1.

The full results of the experiments are listed in Tables 4-26. Table 5 shows the sizes of the sub-and super-concepts sets for weakening when removing wrong axioms one at a time in different orders using Algorithm C2. Tables 6-10 show the sizes of the super- and sub-concepts sets for weakening different ontologies using Algorithms C1-C4.

When the completion is added, we can reduce the amounts of concepts in the completed axiom sets by only showing combinations that would not introduce equivalence between concepts in the ontology. This means that in the implemented version of the completing algorithm, *sp* should belong to $sup(\alpha, \mathcal{T}) \setminus sup(\beta, \mathcal{T})$ ($sup(\alpha, \mathcal{T})$ in the original algorithm) and *sb* to $sub(\beta, \mathcal{T}) \setminus sub(\alpha, \mathcal{T})$ ($sub(\beta, \mathcal{T})$ in the original algorithm). These new sets of super- and sub-concepts are called *source* and *target*. We ran several comparative experiments showing the difference between the sizes of source/target sets and the sizes of $\mathrm{Sup}(\alpha,\mathcal{T})/\mathrm{Sub}(\beta,\mathcal{T})$ sets for the Mini-GALEN ontology and the NCI ontology. Tables 11-18 list the relevant completing results using Algorithms C5-C13. For the remaining ontologies, in order to not introduce equivalence between concepts in the ontology, we only choose the concepts in the source and target sets to generate the completed axioms and Tables 19-26 show the results of the sizes of the source and target sets when completing different ontologies using Algorithms C5-C13.

Table 3. Wrong axioms in each ontology.

Ontology	Wrong axioms
Mini-GALEN	PathologicalProcess⊑InflammationProcess, InflammationProcess⊑GranulomaProcess, Endocarditis⊑PathologicalProcess
PACO	Polish_car⊑Home_improvement_maintenance, Washing_windows⊑Home_improvement_maintenance, Moderate⊑Speed, Washing_car⊑Home_improvement_maintenance, Walking⊑Daily_living_activity, Per_week⊑By_duration
EKAW	Camera_Ready_Paper⊑ ∃writtenBy.Student, Tutorial⊑Conference, Invited_Talk_Abstract⊑Paper, Programme_Brochure⊑Flyer
NCI	Tooth_tissue⊑Tooth, Red_fiber⊑Connective_tissue_fiber, Eye_lid⊑Cheek
Pizza	PineKernels⊑VegetableTopping, PeperoniSausageTopping⊑PeperonataTopping, IceCream⊑ ∃hasTopping.FruitTopping, RosemaryTopping⊑VegetableTopping
OFSMR	Beverage⊑Food, Bread⊑Procesed_fruit_and_vegetables, Pasta⊑Procesed_fruit_and_vegetables

Table 4. Weakening for Mini-GALEN using Algorithms C1-C4. Three wrong axioms give 3 sup/sub-sets per algorithm.

	C1	C2	C3	C4
Sup(β,$\mathcal{T}$) Sub(α,$\mathcal{T}$)	3 2 4 2 3 1	3 2 2 2 1 1	1 2 1 1 1 1	1 2 1 1 1 1
Weakened	PPr ⊑ NPr IPr ⊑ NPr	PPr ⊑ NPr IPr ⊑ NPr	IPr ⊑ NPr	IPr ⊑ NPr

Table 5. Removing wrong axioms in different order for Mini-GALEN by Algorithm C2. Wrong axioms: ①PPr⊆IPr, ②IPr⊆GPr, ③E⊆PPr.

Wrong Axiom	①→②→③	①→③→②	②→①→③	②→③→①	③→②→①	③→①→②
Sup(β,$\mathcal{T}$) Sub(α,$\mathcal{T}$)	3 2 2 2 1 1	3 2 2 2 1 1	3 2 2 2 1 3	2 2 3 1 3 1	2 2 4 1 2 1	3 2 4 1 1 1

Table 6. Weakening the PACO ontology using Algorithms C1-C4. Six wrong axioms give 6 sup/sub-sets per algorithm.

	C1	C2	C3	C4
Sup(β,$\mathcal{T}$) Sub(α,$\mathcal{T}$)	4 4 4 3 4 3 1 1 1 6 1 1	4 4 4 3 4 3 1 1 1 6 1 1	4 4 4 3 4 3 1 1 1 6 1 1	4 4 4 3 4 3 1 1 1 6 1 1

Table 7. Weakening the EKAW ontology using Algorithms C1-C4. Four wrong axioms give 4 sup/sub-sets per algorithm.

	C1	C2	C3	C4
Sup(β,$\mathcal{T}$)	3 4 3 3	3 4 3 3	3 4 3 3	3 4 3 3
Sub(α,$\mathcal{T}$)	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1

Table 8. Weakening the NCI ontology using Algorithms C1-C4. Three wrong axioms give 3 sup/sub-sets per algorithm.

	C1	C2	C3	C4
Sup(β,$\mathcal{T}$)	13 15 8	13 15 8	13 15 8	13 15 8
Sub(α,$\mathcal{T}$)	7 1 3	7 1 3	7 1 3	7 1 3

Table 9. Weakening the Pizza ontology using Algorithms C1-C4. Four wrong axioms give 4 sup/sub-sets per algorithm.

	C1	C2	C3	C4
Sup(β,$\mathcal{T}$)	4 8 4 8	4 8 4 8	4 8 4 8	4 8 4 8
Sub(α,$\mathcal{T}$)	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1

Table 10. Weakening the OFSMR ontology using Algorithms C1-C4. Three wrong axioms give 3 sup/sub-sets per algorithm.

	C1	C2	C3	C4
Sup(β,$\mathcal{T}$)	2 4 4	2 4 4	2 4 4	2 4 4
Sub(α,$\mathcal{T}$)	2 1 1	2 1 1	2 1 1	2 1 1

Table 11. Completing the Mini-GALEN ontology using Algorithms C5-C7.

	C5	C6	C7
Weakened	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr
Source	1 1	1 1	1 1
Target	3 2	3 2	3 4
Completed	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$PPr

Table 12. Completing the Mini-GALEN ontology using Algorithms C5-C7.

	C5	C6	C7
Weakened	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr
Sup(α,$\mathcal{T}$)	1 1	1 1	1 1
Sub(β,$\mathcal{T}$)	3 2	3 4	3 4
Completed	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$NPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$PPr	PPr$\sqsubseteq$NPr, IPr$\sqsubseteq$PPr

Table 13. Completing the Mini-GALEN ontology using Algorithms C8-C13.

	C8	C9	C10	C11	C12	C13
Weakened	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑NPr	IPr⊑NPr	IPr⊑NPr
Source Target	3 2 3 2	1 1 2 2	1 1 2 3	1 1 3 2	1 2	1 2
Completed	GPr⊑IPr, PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑PPr	PPr⊑NPr, IPr⊑NPr	IPr⊑NPr	IPr⊑NPr

Table 14. Completing the Mini-GALEN ontology using Algorithms C8-C13.

	C8	C9	C10	C11	C12	C13
Weakened	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑NPr	IPr⊑NPr	IPr⊑NPr
Sup(α,$\mathcal{T}$) Sub(β,$\mathcal{T}$)	4 3 5 5	1 1 2 2	1 1 2 3	1 1 3 4	1 2	1 2
Completed	GPr⊑IPr, PPr⊑NPr, IPr⊑PPr	PPr⊑NPr, IPr⊑NPr	PPr⊑NPr, IPr⊑PPr	PPr⊑NPr, IPr⊑PPr	IPr⊑NPr	IPr⊑NPr

Table 15. Completing the NCI ontology using Algorithms C5-C9.

	C5	C6	C7	C8	C9
Source	3 1 1	3 1 1	3 1 1	6 14 6	3 1 1
Target	40 2143 83	40 2143 83	40 2136 76	59 2143 83	40 2133 76

Table 16. Completing the NCI ontology using Algorithms C5-C9.

	C5	C6	C7	C8	C9
Sup(α,$\mathcal{T}$)	3 1 1	3 1 1	3 1 1	15 16 9	3 1 1
Sub(β,$\mathcal{T}$)	41 2143 83	41 2143 83	41 2136 76	66 2144 86	41 2133 76

Table 17. Completing the NCI ontology using Algorithms C10-C13.

	C10	C11	C12	C13
Source	3 1 1	3 1 1	3 1 1	3 1 1
Target	40 2136 76	40 2143 83	40 2133 76	40 2133 76

Table 18. Completing the NCI ontology using Algorithms C10-C13.

	C10	C11	C12	C13
Sup(α,$\mathcal{T}$)	3 1 1	3 1 1	3 1 1	3 1 1
Sub(β,$\mathcal{T}$)	41 2136 76	41 2143 83	41 2133 76	41 2133 76

Table 19. Completing the PACO ontology using Algorithms C5-C9.

	C5	C6	C7	C8	C9
Source	1 1 1 1 1 1	1 1 1 1 1 1	1 1 1 1 1 1	2 2 2 2 2 3	1 1 1 1 1 1
Target	59 59 59 171 40 40	59 59 59 171 40 40	59 59 59 171 40 40	59 59 59 171 40 40	51 51 51 168 39 39

Table 20. Completing the PACO ontology using Algorithms C10-C13.

	C10	C11	C12	C13
Source	1 1 1 1 1 1	1 1 1 1 1 1	1 1 1 1 1 1	1 1 1 1 1 1
Target	51 52 53 170 39 40	59 59 59 171 40 40	51 51 51 168 39 39	51 52 53 170 39 40

Table 21. Completing of the EKAW ontology using Algorithms C5-C9.

	C5	C6	C7	C8	C9
Source	9 1 1 1	9 1 1 1	9 1 1 1	10 2 2 2	9 1 1 1
Target	23 17 34 34	23 17 34 34	23 17 34 34	23 17 34 34	23 17 33 33

Table 22. Completing of the EKAW ontology using Algorithms C10-C13.

	C10	C11	C12	C13
Source	9 1 1 1	9 1 1 1	9 1 1 1	9 1 1 1
Target	23 17 33 34	23 17 34 34	23 17 33 33	23 17 33 34

Table 23. Completing by the Pizza ontology using Algorithms C5-C9.

	C5	C6	C7	C8	C9
Source	1 1 3 3	1 1 3 3	1 1 3 3	2 7 4 6	1 1 3 3
Target	50 147 50 50	50 147 50 50	50 147 50 50	50 147 50 50	50 144 48 48

Table 24. Completing the Pizza ontology using Algorithms C10-C13.

	C10	C11	C12	C13
Source	1 1 3 3	1 1 3 3	1 1 3 3	1 1 3 3
Target	49 147 48 50	50 147 50 50	18 144 48 48	48 145 49 50

Table 25. Completing the OFSMR ontology using Algorithms C5-C9.

	C5	C6	C7	C8	C9
Source	1 1 1	1 1 1	1 1 1	2 3 3	1 1 1
Target	125 125 125	125 125 125	125 125 125	125 125 125	123 122 122

Table 26. Completing the OFSMR ontology using Algorithms C10-C13.

	C10	C11	C12	C13
Source	1 1 1	1 1 1	1 1 1	1 1 1
Target	123 122 123	125 125 125	123 122 122	123 122 123

F. Implemented systems

We have implemented two systems. As Protégé is a well-known ontology development tool, we implemented a plugin for repairing based on Algorithm C9. Using this algorithm the user can repair all wrong axioms at once. However, by iteratively invoking this plugin the user can also repair the wrong axioms one at a time. Further, we extended the $\mathcal{EL}$ version of the RepOSE system (Wei-Kleiner, Dragisic, and Lambrix 2014; Lambrix, Wei-Kleiner, and Dragisic 2015). We allow the user to choose different combinations, thereby giving a choice in the trade-off between validation work and completeness.

The systems and user manual with examples are available at `https://figshare.com/s/f3b9472a7e5dd69237dc`.

G. Derivation of the Hasse diagrams

For a given TBox $\mathcal{T}$, let Der($\mathcal{T}$) denote the set of derivable axioms from $\mathcal{T}$. Then, for TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$, if $\mathcal{T}_1 \sqsubseteq \mathcal{T}_2$, then we know that Der($\mathcal{T}_1$) $\sqsubseteq$ Der($\mathcal{T}_2$). This means that if an axiom is derivable from TBox $\mathcal{T}_1$, it is also derivable from TBox $\mathcal{T}_2$ (but not necessarily the other way around). As the sub- and super-concepts of a concept are computed using subsumption axioms, this also means that the set of sub-concepts for a concept in $\mathcal{T}_1$ is a subset of the set of sub-concepts for that concept in $\mathcal{T}_2$, and the set of super-concepts for a concept in $\mathcal{T}_1$ is a subset of the set of super-concepts for that concept in $\mathcal{T}_2$. When computing weakened axiom sets and completed axiom sets, the algorithms compute sets of sub-concepts and sets of super-concepts to generate candidate axioms for these weakened and completed axiom sets. Therefore, if $\mathcal{T}_1 \sqsubseteq \mathcal{T}_2$, the sets of candidate axioms for the weakened and completed axiom sets computed for $\mathcal{T}_1$ are subsets of those computed for $\mathcal{T}_2$. This means more validation work for $\mathcal{T}_2$, but also possibly a more complete final ontology. The Hasse diagrams are based on this observation.

Removing. In general, when removing all axioms at once, the TBox is a subset of the TBox with one axiom removed, which in turn is a subset of the TBox where no axioms are removed. When adding no axioms back, the TBox is a subset of the TBox with one axiom added back, which in turn is a subset of the TBox where all axioms are added back. If no wrong axioms are removed, then nothing needs to be added back and thus AB-one, AB-all and AB-none have the same result ($\mathcal{T}_{R-none,AB-all} = \mathcal{T}_{R-none,AB-one} = \mathcal{T}_{R-none,AB-none}$). The TBox for these strategies is larger during computation (of weakened or completed axiom sets) than the TBoxes where one or all wrong axioms are removed. If one wrong axiom at the time is removed, the adding back all (AB-all) or one (AB-one) give the same result ($\mathcal{T}_{R-one,AB-all} = \mathcal{T}_{R-one,AB-one}$) as both strategies add the same one axiom back. The TBox for these strategies is larger than when no wrong axiom is added back ($\mathcal{T}_{R-one,AB-none} \sqsubseteq \mathcal{T}_{R-one,AB-all} = \mathcal{T}_{R-one,AB-one}$). When all wrong axioms are removed at once, then they will be added back at the end or not.[1] However, this does not influence the TBox during the computation. Therefore, the add back strategy does not matter and the TBox during computation is smaller than when wrong axioms were removed one at a time ($\mathcal{T}_{R-all,AB-all} = \mathcal{T}_{R-all,AB-one} = \mathcal{T}_{R-all,AB-none} \sqsubseteq \mathcal{T}_{R-one,AB-none}$).

Weakening. First, we note that updating immediately or updating after each wrong axiom is the same operation for weakening, as a complete weakened axiom set for a wrong axiom is computed. Thus, the TBox for ($\mathcal{T}_{W-one,U-now}$) is the same as for ($\mathcal{T}_{W-one,U-end\_one}$), and the TBox for ($\mathcal{T}_{W-all,U-now}$) is the same as for ($\mathcal{T}_{W-all,U-end\_one}$). Further, when weakening one axiom at a time and updating the TBox (i.e., adding the axioms of the weakened axiom set for a wrong axiom) immediately, results in a larger TBox for the next computations of weakened axiom sets for wrong axioms, than if we would not update immediately ($\mathcal{T}_{W-one,U-end\_all} \sqsubseteq \mathcal{T}_{W-one,U-now}$). When not immediately updating, the TBox for generating the weakened axioms sets stays the same for all wrong axioms and thus gives the same result as weakening all wrong axioms at once. Thus, $\mathcal{T}_{W-all,U-now} = \mathcal{T}_{W-all,U-end\_all} = \mathcal{T}_{W-one,U-end\_all}$.

Completing. When completing one axiom at a time and updating the TBox (i.e., adding the axioms of the completed axiom set for a weakened axiom) immediately, results in a larger TBox for the next computations of completed axiom sets for weakened axioms than not updating immediately ($\mathcal{T}_{C-one,U-end\_one} \sqsubseteq \mathcal{T}_{C-one,C-now}$, $\mathcal{T}_{C-one,U-end\_all} \sqsubseteq \mathcal{T}_{C-one,C-now}$,). When not updating immediately, there is the choice between updating after all weakened axioms for a particular wrong axiom have been processed or waiting until all weakened axioms for all wrong axioms are processed. The TBox for the former case is larger than the one for the latter case ($\mathcal{T}_{C-one,U-end\_all} \sqsubseteq \mathcal{T}_{C-one,U-end\_one}$). Waiting to update the TBox until all weakened axioms for all wrong axioms are processed, means the TBox stays the same during the computation of the completed axioms sets and thus gives the same result as completing all weakened axioms at once ($\mathcal{T}_{C-one,U-end\_all} = \mathcal{T}_{C-all,U-end\_all} = \mathcal{T}_{C-all,U-end\_one} = \mathcal{T}_{C-all,U-now}$).

[1] After completing they should be removed, but after weakening they could be added back for the completion step.