RepOSE–An ontology repairing tool User Guide

Introduction

The RepOSE ontology repairing system is a tool for repairing wrong axioms in an ontology. This tool uses an interactive way which contains the removing, weakening and completing operations to preserve correct knowledge as much as possible to mitigate the influence caused by removing wrong axioms. Currently, we have implemented two versions of this repairing tool. As Protégé is a well-known ontology development tool, we implemented a plugin for repairing based on the Algorithm C9 in the paper[1]. 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. We allow the user to choose different algorithms, representing different ways to combine removing, weakening, completing and updating strategies, thereby providing a choice in the trade-off between validation work and completeness of the repaired ontology.

Contents

---

[1] Proper reference will be given after the review.

1 Protégé plugin

The Protégé plugin is Java-based and uses the OWL API[2], the Protégé package for accessing and handling OWL ontologies and the Protégé editor. It is installed by copying the jar file (ontology repairing) into the plugin directory of the existing main Protégé folder. Then upon a new start, Protégé will find the tool and create a new tab for it. The tool is then found in the tools menu as *Ontology Repairing* (Figure 1).

Below, the general process of repairing is shown together with an example.

1. After loading the target ontology into Protégé and starting the built-in reasoner, the user can click on *Ontology Repairing* in the tools menu to start the guide for repairing.
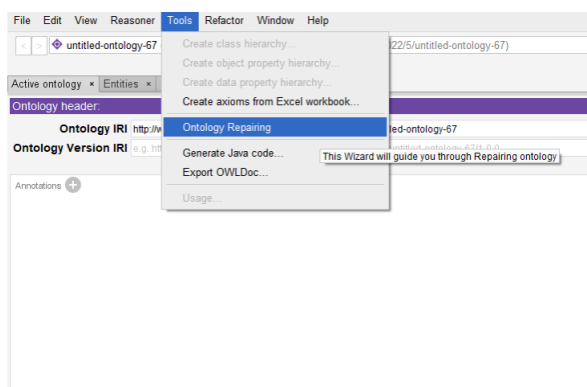

Figure 1: *Ontology Repairing* in the tools menu.

2. First, the user is asked to enter the wrong axiom(s). This is done by selecting left/right-hand concepts from the list of concepts in the ontology and click the *Add axiom* button to add it in the system. (The user can choose to repair the wrong axioms one at a time by iteratively invoking this plugin or input the whole wrong axiom set at once.) As an additional help, when typing the name of a concept, information about this concept, such as disjoint concepts, super-concepts and sub-concepts, is shown. Then, click the *Continue* button to start the weakening step.

Example. Load the Mini-Galen ontology. The wrong axioms are PathologicalProcess ⊑ InflammationProcess, InflammationProcess ⊑ GranulomaProcess, and Endocarditis ⊑ PathologicalProcess. In Figure 2 the axioms PathologicalProcess ⊑ InflammationProcess, InflammationProcess ⊑ GranulomaProcess have already been added. We now choose Endocarditis for the left hand concept and PathologicalProcess for the right hand concept, and click *Add axiom*.

---

[2] http://owlapi.sourceforge.net/

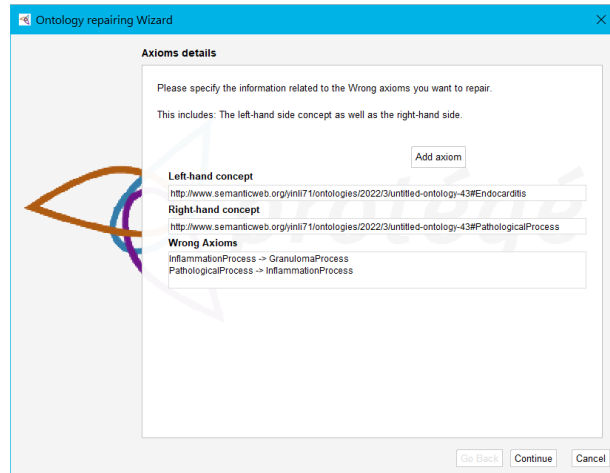This adds Endocarditis $\sqsubseteq$ PathologicalProcess to the set of wrong axioms.

Figure 2: Input the whole wrong axioms set.

3. In the weakening step, the sup/sub concept sets for each wrong axiom are generated. The candidate weakened axioms are axioms $A \sqsubseteq B$ with A in sub and B in sup. These are visualized in two ways: (i) as a list of axioms and (ii) by the sub and sup sets. In the former case weakened axioms are chosen by clicking the *Validate relations* button, selecting the axioms in the list and clicking the *Validate* button. In the latter case the user can choose weakened axioms by clicking on a concept in the sub set and a concept in the sup set and validate the axiom by clicking the *Validate* button. The *Remove wrong* button removes all wrong axioms from the ontology. Click the *Weakening done* button to finish the weakening step and the *Continue* button to go to the completion step.

Example. Figure 3 shows the sub and sup set of the wrong axiom Endocarditis $\sqsubseteq$ PathologicalProcess. There is actually no correct weakened axiom for this wrong axiom, so the user can choose removing this axiom directly by clicking the *remove this axiom directly* button in the axioms list window. Another way is to ignore the axiom, while making sure no incorrect axiom is chosen in the axiom list, i.e., all axioms in the axiom list should be unchecked. Figure 4 shows the sub and sup set of the wrong axiom PathologicalProcess $\sqsubseteq$ InflammationProcess. The weaker axiom PathologicalProcess $\sqsubseteq$ NonNormalProcess is correct and can thus be selected by choosing the PathologicalProcess from the sub set and NonNormalProcess from the super set (visualization) or by choosing from the axiom list as in Figure 5.
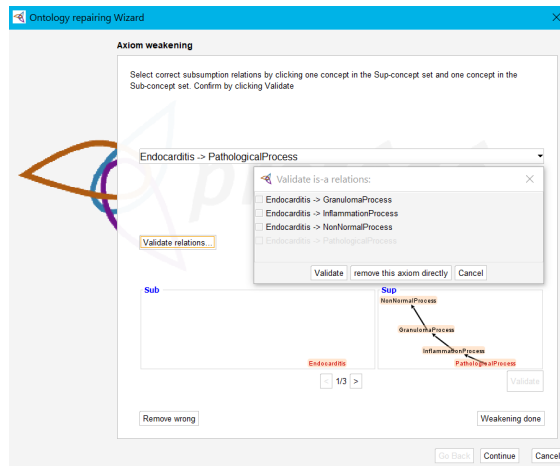
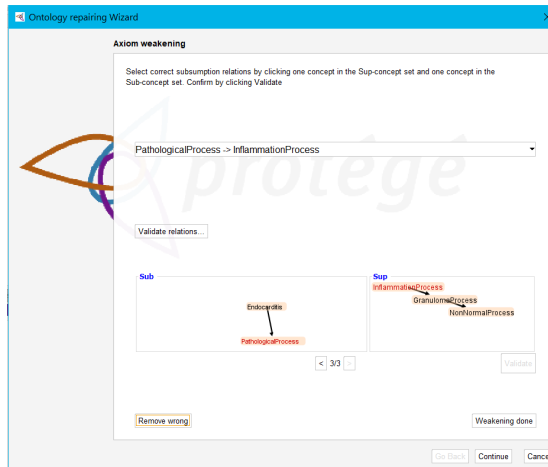Figure 3: The sub- and sup-concepts of Endocarditis ⊑ Pathological-Process .

Figure 4: The sub- and sup-concepts of PathologicalProcess ⊑ Inflammation Process.
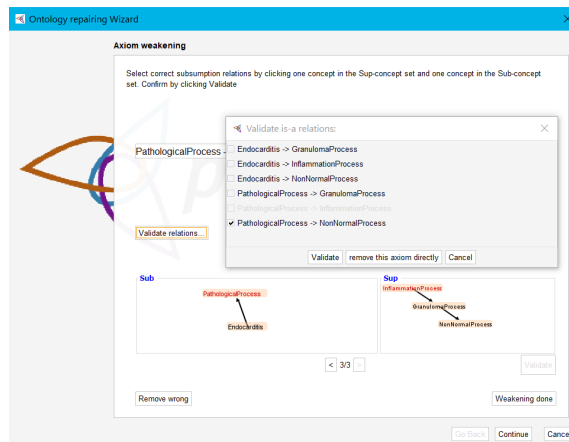

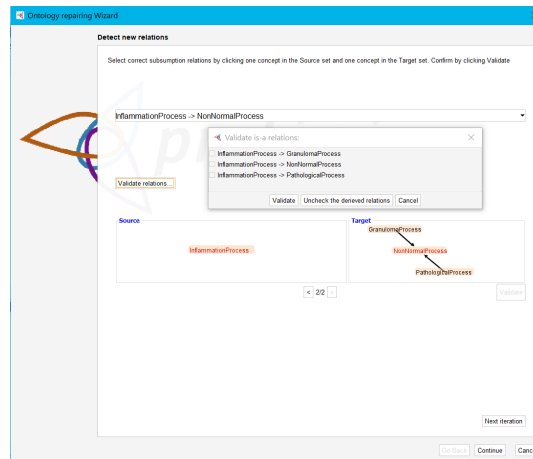Figure 5: The candidate weaker axioms list of PathologicalProcess ⊑ InflammationProcess.

Figure 6: The Source and Target sets of InflammationProcess ⊑ Non-NormalProcess.

4. The procedure in the completion step is similar as in the weakening step. In the completion step, the source and target sets for each axiom are generated. The candidate completing axioms are axioms $A \sqsubseteq B$ with A in source and B in target. These are visualized in two ways: (i) as a list of axioms and (ii) by the source and target sets. In the former case completing axioms are chosen by clicking the *Validate relations* button, selecting the axioms in the list and clicking the *Validate* button. In the latter case the user can choose completing axioms by clicking on a concept in the source set and a concept in the target set and validate the axiom by clicking the *Validate* button. As adding completed axioms adds new knowledge to the ontology that was not earlier derivable, the system allows to find additional correct axioms by invoking the completion process again.[3] This can be done by clicking the *Next iteration* button. To work on the completing for other axioms, click the < and > buttons.

Example. After the weakening step, we obtained the weakened axioms set, which contains PathologicalProcess ⊑ NonNormalProcess and InflammationProcess ⊑ NonNormalProcess. Figure 6 shows the source and target sets of InflammationProcess ⊑ NonNormal-Process. The axiom InflammationProcess ⊑ PathologicalProcess is a correct axiom and was not derivable from the ontology yet. Adding this axiom (by using the visualization, i.e., validating by selecting InflammationProcess from the source and PathologicalProcess from the target, or by using the axiom list) makes the ontology more complete. Similar operations can be performed for the other axiom in the weakened axiom set. In this case no stronger axiom than PathologicalProcess ⊑ NonNormalProcess would be found and thus this axiom is kept as is.

---

[3] The possibility of multiple iterations of the completing phase for a weakened axiom is an extension of the method in the paper.

5. When finishing adding all the axioms, click the *Finish* button at the bottom to close this wizard and the new ontology will be updated automatically. A summary panel is shown. It displays the original wrong axioms, the computed weakened axioms and the completing axioms. The final ontology is created by removing the wrong axioms and adding the completing axioms. The weakened axioms will be derivable from the final ontology.

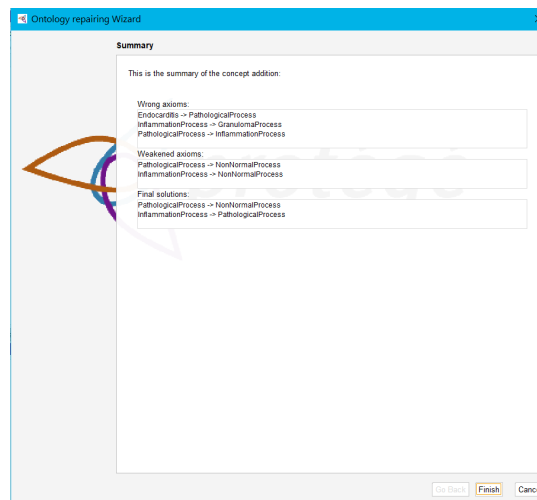**Example.** Figure 7 shows the summary panel for our example.

Figure 7: The summary info panel.

2 The Stand–alone system

We implemented an extended version of the RepOSE system based on the $\mathcal{EL}$ language, which is Java-based system. The general process of repairing is as follows.

1. Click the *Load wrong Ontology* in the file menu to import the target ontology (Figure 8).

Figure 8: Import the target ontology to repair.

2. By clicking the different buttons in the weakening panel, the user can choose different ways to weaken the wrong axioms (Figure 9).

   Weaken All Remove All refers to the combination (R–none,AB–none/W–all,U–end_all) from the paper. It does not remove wrong axioms during the computation, weakens all axioms at once and updates at the end. This is the highest combination for the removing in the Hasse diagrams in the paper and the lowest combination for the weakening.

   Remove All Weaken All refers to the combination (R–all,AB–none/W–all,U–end_all). It removes all wrong axioms before the computation, weakens all axioms at once and updates at the end. This is the lowest combination for the removing in the Hasse diagrams in the paper and the lowest combination for the weakening. It is a choice for the least validation work and the less complete ontologies.

   Remove One Weaken One refers to the combination (R–one,AB–one/W–one,U–end_all). It removes wrong axioms one at a time during the computation and puts them back before dealing with the next wrong axiom. It weakens axioms one at a time and updates at the end.

   For this step *Remove All Weaken All* produces the least validation work and least complete ontologies of the three strategies. *Remove One Weaken One* produces less validation work and less complete ontologies than *Weaken All Remove All*.

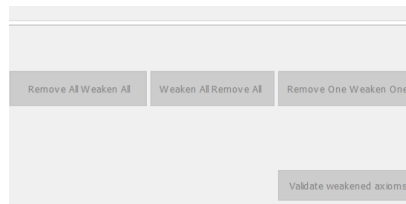   Click the *Generate weaker axioms* (Figure 10) to start the weakening.

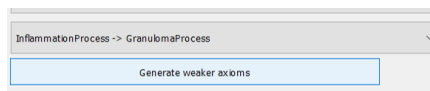Figure 9: Different combinations of weakening and removing.

Figure 10: *Generate weaker axioms* button.

3. In the weakening step, the sup and sub sets for each wrong axiom are generated. The candidate weakened axioms are axioms $A \sqsubseteq B$ with A in sub and B in sup. These are visualized in two ways: (i) as a list of axioms (Figure 11) and (ii) by the sub and sup sets (Figure 12). In the former case weakened axioms are chosen by clicking the *Validate weakened axioms* button, selecting the axioms in the list and clicking the *Validate* button. In the latter case the user can choose weakened axioms by clicking on a concept in the sub set and a concept in the sup set and validate the axiom by clicking the *Validate* button. When no weakened axioms exist for a particular wrong axiom, the user can remove the wrong axiom directly by clicking the *Remove the wrong axiom directly* button.
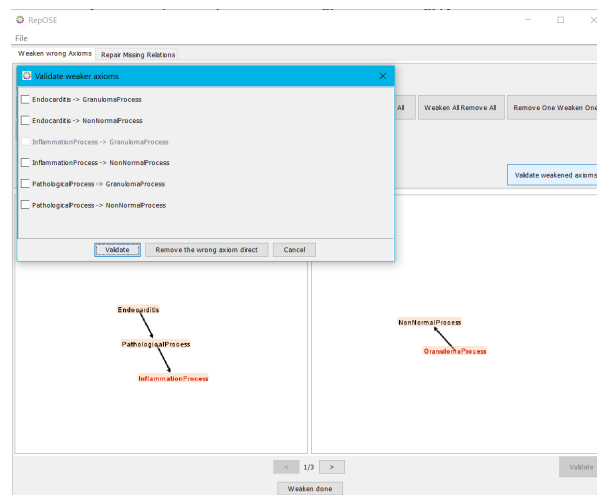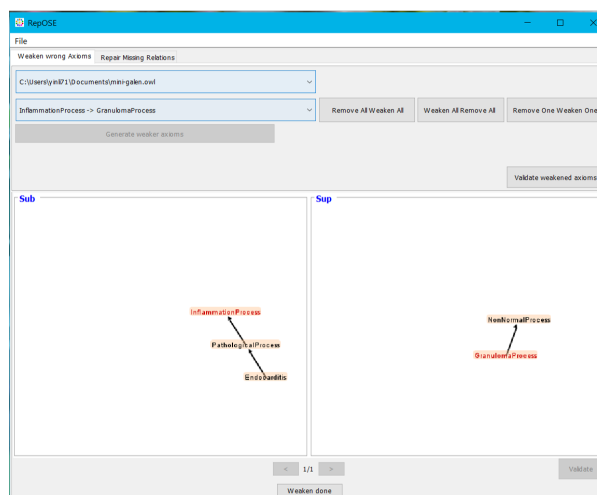

Figure 11: A candidate weakened axioms list.

Figure 12: Visualization of sup and sub sets.

Click the *Weaken done* button to finish the weakening step. The user can save the weakened axioms as well as the current ontology by clicking *Save weakened axioms* and *Save Ontologies and Mappings*
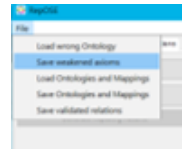
Figure 13: Save weakened axioms.
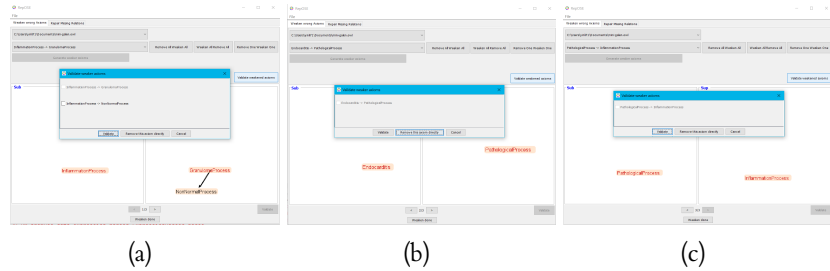

(a)   (b)   (c)

Figure 14: The validation interface for clicking *Remove All Weaken All*.

(Figure 13) in the file menu.

Example. Different sub and sup sets are generated by different algorithms. Choose the desired algorithm by clicking on the corresponding button. Figures 14, 15 and 16 display the validation interfaces when choosing *Remove All Weaken All*, *Weaken All Remove All*, and *Remove One Weaken One* respectively.

Figures 14b, 15b and 16b show the sub and sup sets of the wrong axiom Endocarditis ⊑ PathologicalProcess. There is no correct weakened axiom for this wrong axiom, so the user can either remove this axiom directly by clicking the *Remove the wrong axiom directly* button in the axioms list window or ignore the axiom, while making sure no incorrect axiom is chosen in the axiom list, i.e., all axioms in the axiom list should be unchecked.

Figures 14c, 15c and 16c show the sub and sup set of the wrong axiom PathologicalProcess ⊑ InflammationProcess. The weaker axiom PathologicalProcess ⊑ NonNormalProcess is correct and can thus be selected by choosing PathologicalProcess from the sub set and NonNormalProcess from the super set (visualization) or by choosing from the axiom list.

Note, as discussed, *Remove All Weaken All* produces fewer candidate weakened axioms and thus lower validation work than *Remove One Weaken One* which in turn produces fewer candidate weakened axioms and thus lower validation work than *Weaken All Remove All*. The repaired ontologies for *Weaken All Remove All* may be more complete than those for *Remove One Weaken One* which in turn may be more complete than those for *Remove All Weaken All*.
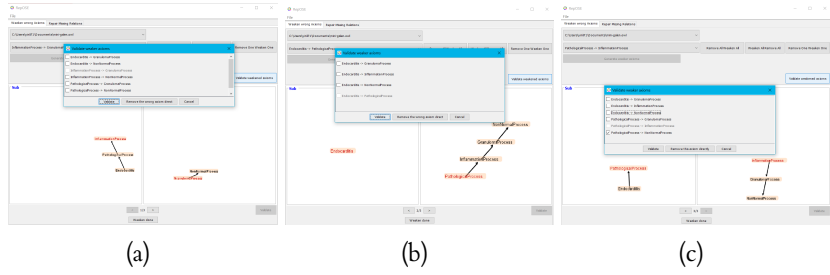
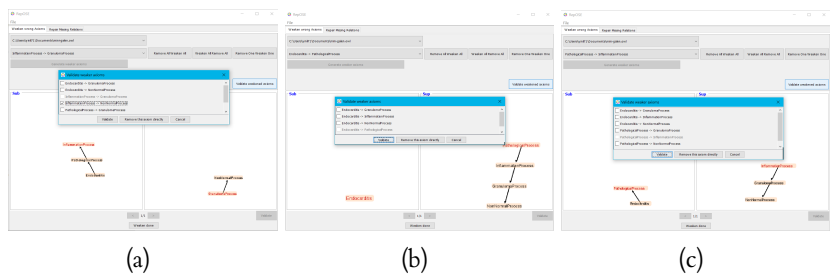Figure 15: The validation interface for clicking *Weaken All Remove All*.

Figure 16: The validation interface for clicking *Remove One Weaken One* .

4. For the completion step, open the system and switch to the *Repair Missing Relations* panel to enter the completing phase (Figure 17). Click the *Load Ontologies and Mappings* in the file menu to import the target ontology.

   The user can choose different ways to complete the ontology by clicking *Complete all* or *Complete one by one.*

   Complete all refers to the combination (C-all,U-end_all) from the paper. It completes all weakened axioms at once and updates at the end.

   Complete one by one refers to the combination (C-one,U-end_one). It completes the weakened axioms one at a time and updates the ontology after the weakened axiom set is handled for each wrong axiom.

   According to the Hasse diagrams in the paper, *Complete one by one* leads to more validation work and more complete ontologies than *Complete all*.

   Then, click *Generate Repairing Actions* button to generate the source and target sets of each weakened axiom.
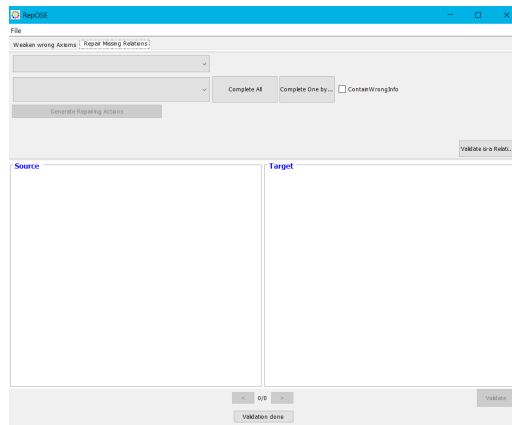

Figure 17: The completion panel.

For the ontology used for completion, the user can choose the ontology that already removed all the wrong axioms or the original one which still contains all the wrong axioms. If the original ontology is used for generating more axioms, the user needs to check the *WrongInfo* box as Figure 18.
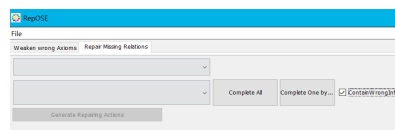

Figure 18: The WrongInfo checkbox.

5. The procedure of completion is similar as in the weakening phase. Based on the generated source and target sets, the candidate completing axioms can be validated through selecting axioms from the axiom list when clicking the *Validate is-a Relations* button in the panel, or by clicking on a concept in the source set and a concept in the target set and then click the *Validate* button at the bottom.

Example. For the example we use the *Complete all* strategy. Further, we use the the weakened axiom set computed by choosing *Weaken All Remove All* in the weakening phase: { PathologicalProcess ⊑ NonNormalProcess, InflammationProcess ⊑ NonNormalProcess}. (Figure 19 shows how we have chosen these axioms from the candidate weakened axioms list during the weakening step.). Figures 20 and 21 show the source and target sets generated by using the ontology where the wrong axioms are removed and the original ontology, respectively.
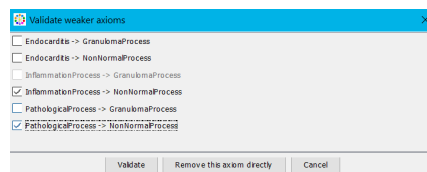

Figure 19: The weakened axiom set computed by choosing *Weaken All Remove All*.

Figure 21 shows candidate completion axioms when the original ontology, which still contains the wrong axioms, is used as TBox. It therefore generates more candidate completion axioms than when the wrong axioms are removed (Figure 20). An axiom that is correct and that was not derivable from the ontology yet is, for instance, InflammationProcess ⊑ PathologicalProcess in Figure 21a,b. Adding this axiom (by using the visualization, i.e., validating by selecting InflammationProcess from the source and PathologicalProcess from the target, or by using the axiom list) makes the ontology more complete. Similarly for the axiom GranulomaProcess ⊑ InflammationProcess. Similar operations can be performed for the other axiom in the weakened axiom set.

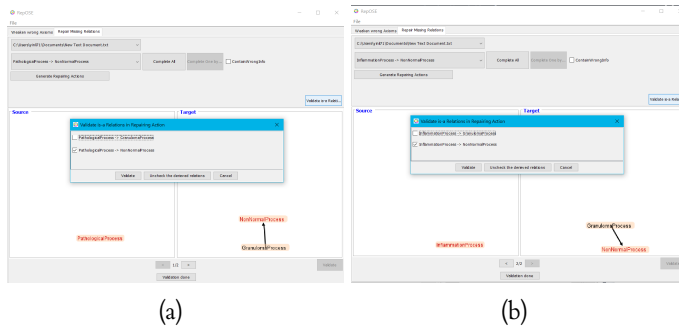(a)   (b)

Figure 20: The validation interface when using the ontology where
the wrong axioms are removed.

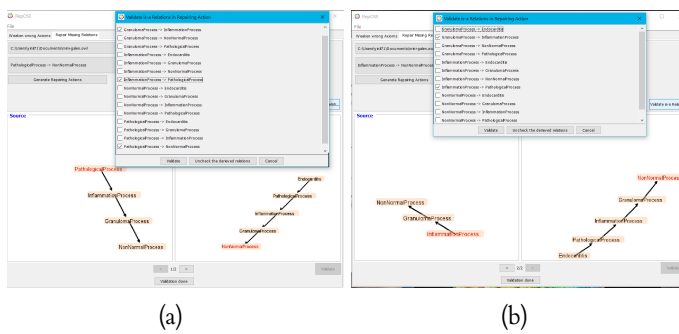

(a)   (b)

Figure 21: The validation interface when using the original ontology.

6. When all the validations are finished, click the *Validation done* button.
   The user can click *Save Ontologies and Mappings* and *Save validated
   relations* from the file menu to save the new ontology as well as the
   actual axioms that will be added to the ontology.