Instructions for the Protege Annotation Experiment

Thank you for participating in this experiment!

The experiment consists in performing two ontology engineering tasks on a given ontology that has been generated automatically using an ontology learning algorithm. The tasks focus on checking whether the extracted concepts are relevant for the climate change and financial domains (**Class Relevance Check** task) and then on verifying whether the extracted subsumption (isA) relations are correct (**SubClass Check** task). Operate at your normal work pace and please complete the tasks for **all** concepts and (subClassOf-)relations in the ontology.

For each task, please write down the start and end time of your work on that task, as the duration of your work is an important aspect for our research.

You can use any additional resources during this experiment if you need additional information - for example, if you are not sure about the meaning of a concept, you can retrieve this information from the Web.

Remarks:

- You have to state the relevance / subClass relation only once in the ontology -- even if the concept appears at multiple points in the ontology.
- Underscores ("_") in concept names are OK, we use them to represent the "space" character.

1. Class Relevance Check Task

Start:..... End:.....

The goal of this task is to verify whether the classes of the given ontology are relevant for the climate change domain (for example, if you would expect to see such a term in a news article about climate change but not about other topics).

Details:

1. Open the **"climate_change_class_relevance.rdf"** ontology in Protege.
2. Go into class view.
3. For all ontology classes (please don't forget any class) do the following:
    a. Think about: Is this class relevant for the given domain of "climate change" in your opinion.
    b. Use the boolean annotation "uComp_class_relevance" to annotate every **class** in the ontology. The annotation (type) already exists, just select it from the list of available annotations. *A remark: You can add annotations in the righthand (upper) frame of Protege, just select the Annotation tab and click on the + symbol.*

c. If the class is not relevant, because it is too generic, too specific or “non-sense”, set the “uComp_class_relevance” annotation to false. A class is also not relevant if it is not an english term (if it contains a typo, etc). Simply enter “false” as a value for this annotation. When you enter the first annotation, you also have to select the annotation type (boolean) from the list.
d. If the class is relevant, set the “uComp_class_relevance” annotation to relevant by providing the annotation value “true”.

4. When you are done with all classes, save the result. We also recommend that you frequently save the ontology during your work session.
5. Proceed to the “Subclass Check” task.

2. SubClass Check task
Start:..... End:.....

The goal of this task is to check whether the subClass relations in the ontology are correct.

Details:

6. Open the ontology **“climate_change_subclassof_check.rdf”** in Protege
7. Go into class view.
8. For each **subClass relation** in the ontology (please don’t forget any) do the following:
    a. Think about: Is there really a subClassOf relation between the two classes for the given domain of “climate change” in your opinion -- and is the direction of the subClassOf relation correct? Advice: Best start from the “lowest” subClass.
        i. Example a: human - subClassOf - mammal → correct
        ii. Example b: mammal - subClassOf human → incorrect

    b. Use the annotation “uComp_subclassof_check” to annotate all **subClassOf** relations in the ontology. The annotation (type) already exists, just select it from the list of available annotations.*A remark: You can add annotations in the righthand (lower) frame of Protege, look für “SubClass Of” and click on the @ symbol.*
    c. If yes, add an annotation “uComp_subclassof_check” with the value “true”. When you enter the first annotation, you also have to select the annotation type (boolean) from the list.
    d. If no, add an annotation “uComp_subclassof_check” with the value “false”.
9. When you are done with all subClass relations, save the result. We also recommend that you frequently save the ontology during your work session.
10. You’re done, tell the supervisor. Thank you, great job!

3. Class Relevance Check Task (Finance Domain)
Start:..... End:.....

Do task 1 also for ontology **“finance_ontology--class_relevance_check.rdf”** in the domain of “Currencies / Euro crisis”.

4. SubClass Check task (Finance Domain)
Start:..... End:.....

Do task 2 also for ontology **“finance_ontology--subclassof_check.rdf”** in the domain of “Currencies / Euro crisis”.