

## Debugging the missing is-a structure of ontologies

## Outline

- n Background
- n Theory of our approach
- n Debugging approach
- n Implemented system
- n Experiments
- n Future Work

2

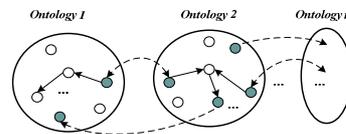
## Outline

- n **Background**
- n Theory of our approach
- n Debugging approach
- n Implemented system
- n Experiments
- n Future Work

3

## Ontology Networks

An **ontology network** consists of a set of **ontologies** and **mappings** between them.



4

## Defects in ontology networks

- n Neither developing ontologies nor finding mappings between ontologies is an easy task.
- n It may happen that
  - n mappings between ontologies are not correct/complete
  - n ontologies are not correct/complete
  - n the integrated ontology network is not consistent

5

## Defects in ontology networks

- n mappings between ontologies are not correct/complete
  - n Ontology alignment
  - n Debugging mappings
- n ontologies are not correct/complete
  - n Ontology debugging
- n the integrated ontology network is not consistent
  - n Ontology network debugging

6

## Defects in ontologies

- n Syntactic defects
  - eg. wrong tags or incorrect format
  - Easy to detect and resolve
- n Semantic defects
  - eg. unsatisfiable concepts or inconsistent ontologies
  - There has been some work on detection
  - In recent years, there is some work on repairing
- n Modeling defects
  - eg. wrong or missing relations
  - Solution requires domain knowledge.
  - There is some work on detection, but little work on repairing

7

## Ontologies

- n Developing ontologies is not an easy task.
  - Many ontologies have some underlying defects.
- n Such ontologies, although often useful, also lead to problems when used in semantically-enabled applications.
  - Wrong conclusions may be derived or valid conclusions may be missed.

8

## Missing is-a relations

- n In 2008 Ontology Alignment Evaluation Initiative (OAEI) Anatomy track, task 4
  - Ontology MA : Adult Mouse Anatomy Dictionary (2744 concepts)
  - Ontology NCI-A : NCI Thesaurus - anatomy (3304 concepts)
  - Partial reference alignment between them (988 mappings)
    - n 121 missing is-a relations in MA
    - n 83 missing is-a relations in NCI-A

9

## Ontologies used ...

- n Ontology-based querying.

PubMed.gov Search: PubMed Limits Advanced search Help  
 "Scleral Diseases" [MeSH] Search Clear  
 Medical Subject Headings (MeSH)  
 All MeSH Categories  
 Diseases Category  
 Eye Diseases  
 Scleral Diseases  
 Scleritis  
 ...  
 return 1363 articles

10

## Influence of Missing Structure

- n Incomplete results from ontology-based queries

PubMed.gov Search: PubMed Limits Advanced search Help  
 "Scleral Diseases" [MeSH] Search Clear  
 Medical Subject Headings (MeSH)  
 All MeSH Categories  
 Diseases Category  
 Eye Diseases  
 Scleral Diseases  
 Scleritis  
 ...  
 return 1363 articles  
 return 613 articles  
 55% results are missed!

11

## Debugging missing is-a structure in ontology networks

Given a set of ontologies networked by a set of **correct** mappings, how to **detect and repair the missing is-a relations in these networked ontologies?**

*Ontology Debugging* – is the process of detecting and repairing the defects within ontologies.

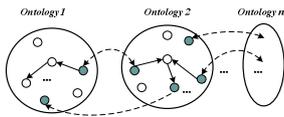
12

## Assumptions

- n We assume that **all the existing mappings** in the ontology network are **correct**.

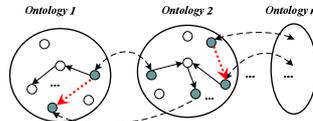
*Partial Reference Alignment (PRA)* – is a set of correct mappings between two ontologies.

- n The existing correct mappings are called **PRA mappings**.
- n Concepts in PRA mappings are called **PRA concepts**.



13

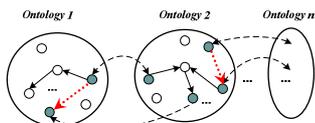
## Scope



- n We focus on ontologies represented as *taxonomies*, which consist of *named concepts* and *is-a relations*.
- n For the PRA mappings, we considered *equivalence* and *subsumption* mappings between *concepts* in the ontologies.

14

## Scope



- n Given an ontology network, we assume that **all the existing structure in ontologies is correct**.

15

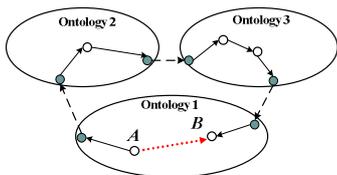
## Outline

- n Background
- n **Theory of our approach**
- n Debugging approach
- n Implemented system
- n Experiments
- n Future Work

16

## Definition of missing is-a relations

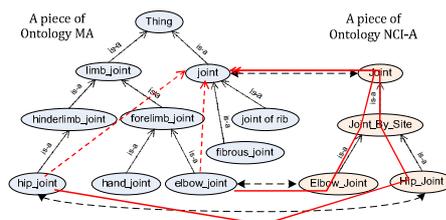
Given two concepts A and B in an ontology O in the network. If "A is-a B" is **logically derivable from the ontology network**, but **not from the ontology O alone**, then "A is-a B" is a **missing is-a relation**.



17

## Example of missing is-a relations

- n Two small pieces of ontologies MA and NCI-A about concept "joint", along with 3 equivalence mappings between them.



18

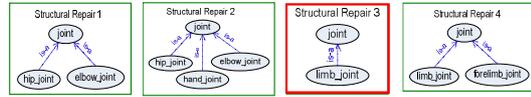
## Repairing missing is-a relations

Repair the original ontologies by adding a set of is-a relations (called **structural repair**) to each ontology, such that the missing is-a relations can be derived from the extended ontology.

- n Structural repair
    - n The is-a relations within the structural repair are called 'repairing actions'.
- The set of missing is-a relations themselves is a structural repair, but it is not always the only nor the best choice.

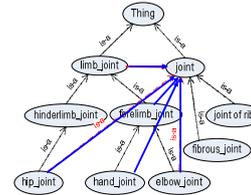
19

## Example



**Question:**  
How can we recognize structural repairs that are interesting for a domain expert?

We defined three repair preferences.

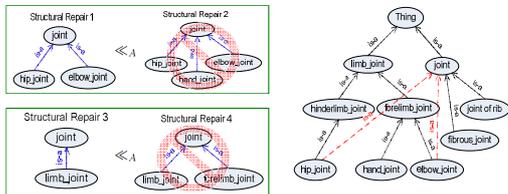


20

## Repair Preference I

- n Axiom-Preference

Prefer to use structural repair **without non-contributing** repairing actions.

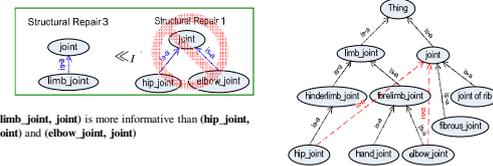


21

## Repair Preference II

- n Information-Preference

Prefer to use structural repair with **more informative** repairing actions.



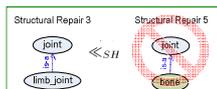
(limb\_joint, joint) is more informative than (hip\_joint, joint) and (elbow\_joint, joint)

22

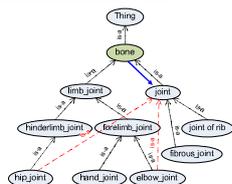
## Repair Preference III

- n Strict-Hierarchy-Preference

Prefer to use the structural repair which **does not change the existing is-a relations in the original ontology into equivalence relations**.



(bone, joint) will introduce an equivalence relation between 'joint' and 'bone'.



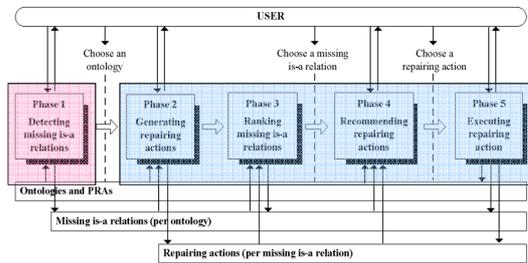
23

## Outline

- n Background
- n Theory of our approach
- n **Debugging approach**
- n Implemented system
- n Experiments
- n Future Work

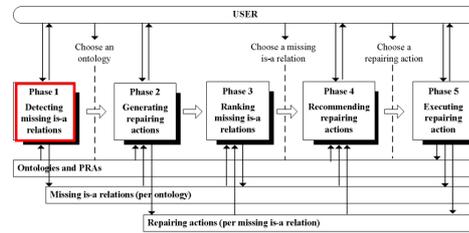
24

## Overview of debugging approach



25

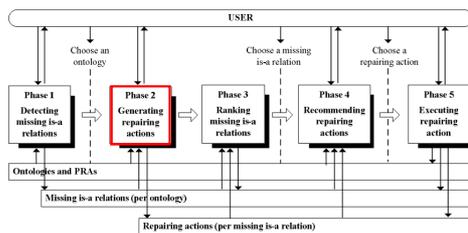
## Phase 1. Detecting missing is-a relations



- The algorithm is based on the definition of missing is-a relations mentioned before.

26

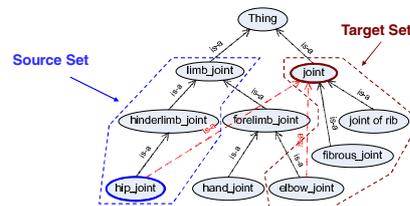
## Phase 2. Generating repairing actions



27

## Example

For missing is-a relation (**hip\_joint**, **joint**), we generate two sets of concepts, which result in 3x4 possible repairing actions.



28

## Algorithm 1 - basic algorithm

### Intuition

- Given a set of missing is-a relations, find possible repairing actions taking into account that all missing is-a relations will be repaired.

**Input**  
The ontology under repair  $O$ , its set of missing is-a relations  $M$ .  
**Output**  
Repairing actions.  
**Algorithm**  
1. Initialize KB with ontology;  
2. For every missing is-a relation  $(a, b) \in M$ : add the axiom  $a \rightarrow b$  to the KB;  
3. For each  $(a, b) \in M$ :  
     $Source(a, b) := \text{super-concepts}(a) - \text{super-concepts}(b)$ ;  
     $Target(a, b) := \text{sub-concepts}(b) - \text{sub-concepts}(a)$ ;  
4. Missing is-a relation  $(a, b)$  can be repaired by choosing an element from  $Source(a, b) \times Target(a, b)$ .

Figure 4.5: The basic algorithm for generating repairing actions.

29

## Algorithm 1 - basic algorithm

### Consistent with the repairing preferences

- For a repairing action  $(s, t)$  regarding missing is-a relation  $(a, b)$ , it is guaranteed that
  - since  $a \rightarrow s$  and  $t \rightarrow b$ 
    - $(s, t)$  is relevant for repairing  $(a, b)$  **Axiom-Preference**
    - $(s, t)$  is more informative than  $(a, b)$  **Information-Preference**
  - $(a, t)$  and  $(s, b)$  will not introduce equivalence relations, where in the original ontology we have only is-a relations **Strict-hierarchy-Preference**

- For each  $(a, b) \in M$ :  
     $Source(a, b) := \text{super-concepts}(a) - \text{super-concepts}(b)$ ;  
     $Target(a, b) := \text{sub-concepts}(b) - \text{sub-concepts}(a)$ ;  
4. Missing is-a relation  $(a, b)$  can be repaired by choosing an element from  $Source(a, b) \times Target(a, b)$ .

30

## Algorithm 2 - extended algorithm

### Intuition:

- Taking into account influence of other missing is-a relations that are common to all possible choices for repairing actions of other missing is-a relations.

**Input**  
The ontology under repair  $O$ , its set of missing is-a relations  $M$ .

**Output**  
Repairing actions.

**Algorithm**

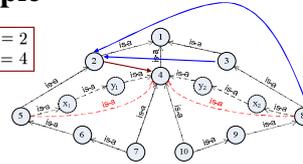
- Initialize KB with ontology;
- For every missing is-a relation  $(a, b) \in M$ :  
Create two new concepts  $x$  and  $y$  in the KB;  
Add the axioms  $a \rightarrow x, x \rightarrow y, y \rightarrow b$  to the KB;
- For each  $(a, b) \in M$ :  
 $Source-ext(a, b) = \text{super-concepts}(a) - \text{super-concepts}(x)$ ;  
 $Target-ext(a, b) = \text{sub-concepts}(b) - \text{sub-concepts}(y)$ ;
- Missing is-a relation  $(a, b)$  can be repaired by choosing an original ontology element from  $Source-ext(a, b)$  and an original ontology element from  $Target-ext(a, b)$ .

Figure 4.7: The extended algorithm for generating repairing actions.

31

## Example

$x_1 = 2$   
 $y_1 = 4$

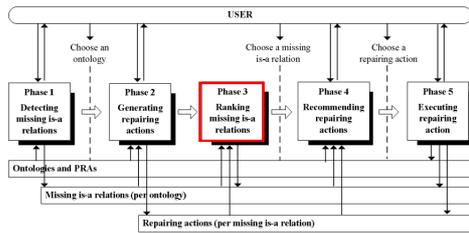


$Source-ext(5, 4) = \{5, 4, 1, 2, x_1, y_1\} - \{4, 1, x_1, y_1\} = \{5, 2\}$   
 $Target-ext(5, 4) = \{4, 8, 9, 10, 5, 6, 7, x_1, y_1, x_2, y_2\} - \{5, 6, 7, x_1, y_1\}$   
 $= \{4, 8, 9, 10, x_2, y_2\}$   
 $Source-ext(8, 4) = \{8, 4, 1, 3, x_2, y_2\} - \{4, 1, x_2, y_2\} = \{8, 3\}$   
 $Target-ext(8, 4) = \{4, 8, 9, 10, 5, 6, 7, x_1, y_1, x_2, y_2\} - \{8, 9, 10, x_2, y_2\}$   
 $= \{4, 5, 6, 7, \boxed{2}\}$

For instance, if we choose repairing action (2,4) for missing is-a relation (5, 4), which means  $x_1$  and  $y_1$  will become equivalent to 2 and 4 respectively, the influence is that concept 2 will become a new element in  $Target-ext(8, 4)$ .

32

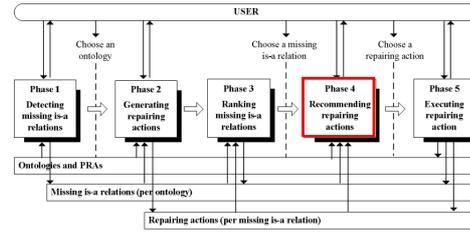
## Phase 3. Ranking missing is-a relations



- Rank the missing is-a relations with respect to the number of possible repairing actions.

33

## Phase 4. Recommending repairing actions



- Recommend repairing actions based on external domain knowledge, such as WordNet and UMLS.

34

## Recommending algorithm

- We assume that we can query the external domain knowledge regarding subsumption of concepts

- General thesauri
  - e.g. WordNet
- Specialized domain-specific sources
  - e.g. UMLS (Unified Medical Language System)

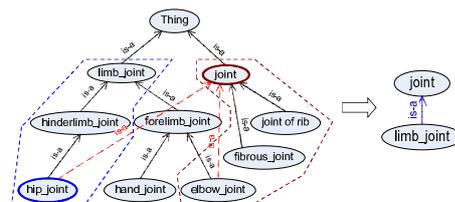
### Algorithm

- Given a missing is-a relation with possible repairing actions, recommend the most informative repairing actions that are supported by evidence in the domain knowledge.

35

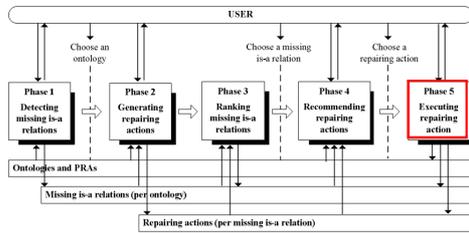
## Example

For missing is-a relation (hip\_joint, joint), we get the recommendation (limb\_joint, joint) from the 12 possible repairing actions.



36

## Phase 5. Executing repairing actions



37

## Executing repairing actions

- n Intuition
  - Every time a repairing action is chosen and executed, the repairing actions for the other missing is-a relations need to be recomputed based on the ontology extended with the chosen repairing action.
  - In order to facilitate updates, we introduce an algorithm to keep track of the influences.

38

## Executing repairing actions

- n Intuition
  - After a repairing action  $(X, Y)$  is executed, for any other missing is-a relations, for example  $(A, B)$ :
    - n **Source** $(A, B)$  changes only when A or B is a sub-concept of X
    - n **Target** $(A, B)$  changes only when A or B is a super-concept of Y

39

## Outline

- n Background
- n Theory of our approach
- n Debugging approach
- n **Implemented system**
- n Experiments
- n Future Work

40

41

## Outline

- n Background
- n Theory of our approach
- n Debugging approach
- n **Implemented system**
- n **Experiments**
- n Future Work

42

## Experiment

- n We have done experiments regarding performance and feasibility of our system.

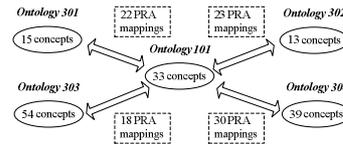
43

## Datasets

- n Anatomy dataset



- n Bibliography dataset



44

## Experiment Result

- n Anatomy Dataset
  - At the beginning, we found 199 missing is-a relations in MA and 167 in NCI-A.
  - During the repairing
    - We found 6 additional missing is-a relations in MA, 10 in NCI-A.
    - For 25 missing is-a relations in MA and 11 in NCI-A, the repairing actions changed.
    - In most cases, the ranking and recommendations seemed useful.
    - Most source and target sets are small enough to allow a good visualization.
  - Extended algorithm: influences for most missing is-a relations; clusters
  - The whole debugging process took about 3 hours.

45

## Experiment Result

- n Bibliography Dataset
  - At the beginning, we found 22 missing is-a relations in ontology 101, 20 in ontology 304, 1 in each of the others.
  - During the repairing
    - We found 3 additional missing is-a relations in ontology 304.
    - The whole debugging process took about 5 minutes.

46

## Experiment Result

- n Recommending repairing actions
  - We use WordNet as domain knowledge.
  - The running time for generating recommendations for all missing is-a relations was
    - Circa 4 minutes for MA
    - Circa 2 minutes for NCI-A
  - Concerning the number of recommendations
    - MA: 19 receive 1; 12 receive 2; 2 receive 3.
    - NCI-A: 5 receive 1.

47

## Experiment Result

- n Anatomy Dataset

	total initially	equivalence initially	redundant initially	to repair initially
MA	199	6	78	115
NCI-A	167	3	84	80

Figure 17: Scenario 1 - Initially detected missing is-a relations.

	total during	equivalence during	redundant during	to repair during
MA	6	0	1	5
NCI-A	10	0	3	7

Figure 29: Scenario 1 - Additionally detected missing is-a relations during whole debugging session.

	total repaired	by others	obvious self	obvious non-self	ask recommendation
MA	120	101	19	28	0
NCI-A	87	87	0	7	0

Figure 25: Scenario 1 - Repaired missing is-a relations.

	total use rec self	use rec non-self	not use rec self	not use rec non-self
MA	73	52	16	3
NCI-A	80	73	6	0

Figure 28: Scenario 1 - Recommendations.

48

## Outline

- n Background
- n Theory of our approach
- n Repairing the structure of an ontology
- n Implemented system
- n Experiments
- n **Future Work**

49

## Future work

- n Debugging is-a Structure within Networked Ontologies
  - Presence of wrong mappings between ontologies, or wrong is-a structure in ontologies
  - Consider more expressive ontologies (e.g. knowledge base)
- n Investigate the interaction and integration of ontology alignment and ontology debugging process.

50

51