

Ontology Integration Using Mappings: Towards Getting the Right Logical Consequences

*Ernesto Jiménez Ruiz*¹ Bernardo Cuenca Grau²
Ian Horrocks² Rafael Berlanga¹

¹Computer Languages and Systems, Universitat Jaume I, Spain

²Computing Laboratory, University of Oxford, UK

ESWC 2009 Crete June 3, 2009

Outline

- 1 Introduction
- 2 Ontology Integration Method
- 3 Method Step 1: Mapping Computation
- 4 Method Step 2: Computation of Entailments
- 5 Method Step 3: Evaluation of Entailments
- 6 Method Step 4: Repair of Unintended Entailments
- 7 Related Work
- 8 Evaluation and Conclusions

Our approach in a nutshell

Logic-based Ontology Integration using Mappings

- We present a methodology and tool support
- We evaluate the consequences of integrating ontologies

Tool support — Protégé plugin **ContentMap**

- Operative for Protégé 4 beta

Our approach in a nutshell

Logic-based Ontology Integration using Mappings

- We present a methodology and tool support
- We evaluate the consequences of integrating ontologies

Tool support — Protégé plugin **ContentMap**

- Operative for Protégé 4 beta

Why are Ontologies Integrated?

Some Reasons...

- Ontology reuse in ontology development
- Data integration
- Interoperability between agents

Some Problems...

- How to establish mappings between ontologies
- How to evaluate compatibility between ontologies

Motivating Example

Bibliographic domain ...

- Ontologies from the 2004 EON Ontology Alignment Contest.
- Describing bibliographic references: INRIA (\mathcal{O}_{INR}), MIT (\mathcal{O}_{MIT}), UMBC ($\mathcal{O}_{\text{UMBC}}$) and AIFB Karlsruhe ($\mathcal{O}_{\text{AIFB}}$).
- \mathcal{O}_{INR} was used as reference.
- We use the corresponding **gold standard** to evaluate the integration.

Motivating Example: Integrating \mathcal{O}_{MIT} and \mathcal{O}_{INR}

Some Mappings ...

- $\mathcal{O}_{INR} : \text{year} \sqsubseteq \mathcal{O}_{MIT} : \text{hasYear}$
- $\mathcal{O}_{INR} : \text{TechnicalReport} \sqsubseteq \mathcal{O}_{MIT} : \text{Technicalreport}$

Unintended Consequences ...

- $\text{TechnicalReport} \sqsubseteq \text{Date}$
- $\text{TechnicalReport} \sqsubseteq \exists \text{date.Reference}$

Explanation

- In \mathcal{O}_{MIT} : $\text{TechnicalReport} \sqsubseteq \geq \text{hasYear } 1.\text{Literal}$
- In \mathcal{O}_{INR} : $\text{year} \text{ hasDomain } \text{Date}$

Motivating Example: Integrating \mathcal{O}_{MIT} and \mathcal{O}_{INR}

Some Mappings ...

- $\mathcal{O}_{\text{INR}} : \text{year} \sqsubseteq \mathcal{O}_{\text{MIT}} : \text{hasYear}$
- $\mathcal{O}_{\text{INR}} : \text{TechnicalReport} \sqsubseteq \mathcal{O}_{\text{MIT}} : \text{Technicalreport}$

Unintended Consequences ...

- $\text{TechnicalReport} \sqsubseteq \text{Date}$
- $\text{TechnicalReport} \sqsubseteq \exists \text{date.Reference}$

Explanation

- In \mathcal{O}_{MIT} : $\text{TechnicalReport} \sqsubseteq \geq \text{hasYear } 1.\text{Literal}$
- In \mathcal{O}_{INR} : $\text{year} \text{ hasDomain } \text{Date}$

Motivating Example: Integrating \mathcal{O}_{MIT} and \mathcal{O}_{INR}

Some Mappings ...

- $\mathcal{O}_{INR} : \text{year} \sqsubseteq \mathcal{O}_{MIT} : \text{hasYear}$
- $\mathcal{O}_{INR} : \text{TechnicalReport} \sqsubseteq \mathcal{O}_{MIT} : \text{Technicalreport}$

Unintended Consequences ...

- $\text{TechnicalReport} \sqsubseteq \text{Date}$
- $\text{TechnicalReport} \sqsubseteq \exists \text{date.Reference}$

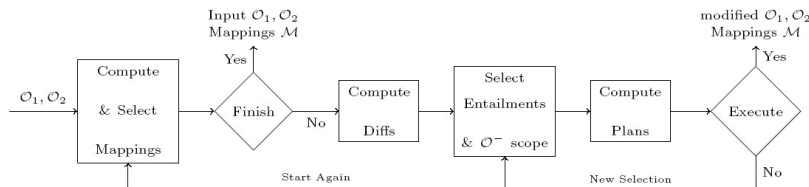
Explanation

- In \mathcal{O}_{MIT} : $\text{TechnicalReport} \sqsubseteq \geq \text{hasYear } 1.\text{Literal}$
- In \mathcal{O}_{INR} : $\text{year hasDomain Date}$

Outline

- 1 Introduction
- 2 Ontology Integration Method**
- 3 Method Step 1: Mapping Computation
- 4 Method Step 2: Computation of Entailments
- 5 Method Step 3: Evaluation of Entailments
- 6 Method Step 4: Repair of Unintended Entailments
- 7 Related Work
- 8 Evaluation and Conclusions

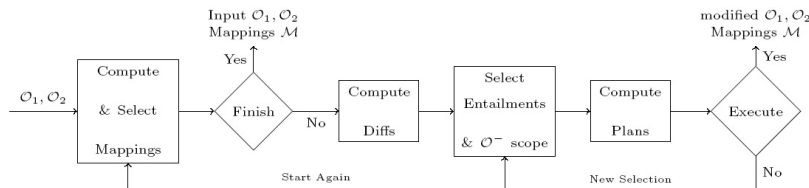
Necessity of a Logic-based Ontology Integration Method



Steps

- Generate or **reuse mappings** between ontologies
- Select and **filter** erroneous mappings
- Reasoning with the ontologies and the mappings
- Compare entailments before and after the integration
- Suggest possible ways to repair errors

Necessity of a Logic-based Ontology Integration Method



Steps

- Generate or **reuse mappings** between ontologies
- Select and **filter** erroneous mappings
- **Reasoning** with the ontologies and the mappings
- **Compare entailments** before and after the integration
- Suggest possible ways to **repair errors**

Tool Support: ContentMap Overview

JIA1.owl (http://www.semanticweb.org/ontologies/2008/9/JIA1.owl) - [C:\AlignmentTools\OntologyTests\JIA_1.owl]

File Edit Ontologies Reasoner Tools Refactor Tabs View Window Help

JIA1.owl (http://www.semanticweb.org/ontologies/2008/9/JIA1.owl)

Active Ontology Entities Classes Object Properties Data Properties Individuals OWL/Viz DL Query ContentMap Manager

ContentMap Manager: [Icons]

Manager Explicit Mappings New Entailments

EXPLICIT MAPPINGS

Input Ontologies

Ontology URI 1 Use Local Loaded Ontology
file:C:/AlignmentTools/OntologyTests/JIA_1.owl

Ontology URI 2 file:C:/AlignmentTools/OntologyTests/JIA_2.owl

Input Mappings

Use pre-extracted mappings file
file:C:/AlignmentTools/OntologyTests/OLA_JIA_mappings.owl

Combine mappings files

Weight 0.0

Weight 0.0

Weight 0.0

Threshold for Suppressible Mappings 0.4

Show Explicit Mappings

IMPACT OF MERGING REFINEMENT

Impact Expressivity Language

'A \sqsubseteq B' (reasoner output)

'A $\sqsubseteq \exists R B$ ' (Existential Restrictions)

'A $\sqsubseteq \forall R B$ ' (Universal Restrictions)

'A $\sqsubseteq \neg B$ ' (disjoint(A,B))
(Being 'A' and 'B' atomic concepts)

Preview Logic Impact Extract Dependency Tree

Suggestion Options

Without Dependency Tree (Mark as suppressible 'A \sqsubseteq B' entailments)

With Dependency Tree (Mark 'A \sqsubseteq B' and low confidence entailments)

Threshold Suppress (<) 0.4 Threshold Keep (==) 0.8

Perform Suggestions

Plan Extraction Options

Select Scope for Plan Extractor:

Allow changes over Mappings

Allow changes over Ontology 1

Allow changes over Ontology 2

Select Order type for Plans:

Order by Number of Axioms

Order by Confidence Value

Extract Plans

Outline

- 1 Introduction
- 2 Ontology Integration Method
- 3 Method Step 1: Mapping Computation**
- 4 Method Step 2: Computation of Entailments
- 5 Method Step 3: Evaluation of Entailments
- 6 Method Step 4: Repair of Unintended Entailments
- 7 Related Work
- 8 Evaluation and Conclusions

Computation of the Mappings

Mapping Representation

- Mappings $\langle id, e_1, e_2, n, \rho \rangle$
- are considered as OWL 2 axioms: $\text{SubClassOf}(e_1 \ e_2)$, $\text{EquivalentClasses}(e_1 \ e_2)$, or $\text{DisjointClasses}(e_1 \ e_2)$
- with ρ of the form (\sqsubseteq) , (\equiv) , or (\perp)
- and n (the confidence value) added as OWL 2 axiom annotation
- **No extra semantics** is given to mappings
- Therefore a set of **mappings** is represented as an **OWL 2 ontology**

Computation of the Mappings

Mapping Representation

- Mappings $\langle id, e_1, e_2, n, \rho \rangle$
- are considered as OWL 2 axioms: `SubClassOf(e1 e2)`, `EquivalentClasses(e1 e2)`, or `DisjointClasses(e1 e2)`
- with ρ of the form (\sqsubseteq) , (\equiv) , or (\perp)
- and n (the confidence value) added as OWL 2 axiom annotation
- **No extra semantics** is given to mappings
- Therefore a set of **mappings** is represented as an **OWL 2 ontology**

Mappings Selection: ContentMap support

Ontology Mapping Tools

- We reuse mapping generated by ontology matching tools
- For the experiments we used OLA, CIDER and AROMA

The image displays two side-by-side screenshots of ontology mapping tool outputs, showing suggested mappings between JIA1 and JIA2 ontologies.

Left Screenshot: EXPLICIT MAPPINGS (O1 → O2)

Suggested Mappings to be Kept:

- effects: JIA1:effects subPropertyOf JIA2:effects (c: 0.8667) ✓
- Negative_Factor: JIA1:Negative_Factor subClassOf JIA2:Positive_Rheum_Factor (c: 0.625) ✓
- JIA1:Negative_Factor subClassOf JIA2:Negative_Rheum_Factor (c: 0.75) ✓
- Disease: JIA1:Disease subClassOf JIA2:Juv_Disease (c: 0.625) ✓
- JIA1:Disease subClassOf JIA2:Disease (c: 1.0) ✓

Right Screenshot: EXPLICIT MAPPINGS (O2 → O1)

Suggested Mappings to be Kept:

- effects: JIA2:effects subPropertyOf JIA1:effects (c: 0.8667) ✓
- Negative_Rheum_Factor: JIA2:Negative_Rheum_Factor subClassOf JIA1:Negative_Factor (c: 0.75) ✓
- JIA2:Negative_Rheum_Factor subClassOf JIA1:Positive_Factor (c: 0.625) ✓
- Positive_Rheum_Factor: JIA2:Positive_Rheum_Factor subClassOf JIA1:Positive_Factor (c: 0.75) ✓
- JIA2:Positive_Rheum_Factor subClassOf JIA1:Negative_Factor (c: 0.625) ✓

Outline

- 1 Introduction
- 2 Ontology Integration Method
- 3 Method Step 1: Mapping Computation
- 4 Method Step 2: Computation of Entailments**
- 5 Method Step 3: Evaluation of Entailments
- 6 Method Step 4: Repair of Unintended Entailments
- 7 Related Work
- 8 Evaluation and Conclusions

Computation of New Entailments

Reasoning with \mathcal{M} , \mathcal{O}_1 and \mathcal{O}_2

- $\mathcal{M} \subseteq \text{map}(\mathcal{O}_1, \mathcal{O}_2)$ given by a tool
- $\mathcal{U} := \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$
- New entailments in \mathcal{U} but not in \mathcal{O}_1 , \mathcal{O}_2 or \mathcal{M} , regarding their respective signatures.

Computation of New Entailments: *Semantic Difference*

How to extract new entailments in \mathcal{U}

- Notion of *deductive difference*
- B. Konev, D. Walther and Frank Wolter: “*The Logical Difference Problem for Description Logic Terminologies*”

Definition (Deductive Difference)

$$\text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}') = \{\alpha \mid \alpha \text{ a } \mathcal{DL}\text{-axiom, } \mathcal{O} \not\models \alpha, \mathcal{O}' \models \alpha \text{ and } \text{Sig}(\alpha) \subseteq \Sigma\}$$

Definition (Deductive Difference for Mappings)

$$\text{mdiff}_{\Sigma_1, \Sigma_2}(\mathcal{O}, \mathcal{O}') = \{\alpha \in \text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}') \mid \alpha \text{ } \mathcal{DL}\text{-mapping between } \Sigma_1, \Sigma_2\}$$

Computation of New Entailments: *Semantic Difference*

How to extract new entailments in \mathcal{U}

- Notion of *deductive difference*
- B. Konev, D. Walther and Frank Wolter: “*The Logical Difference Problem for Description Logic Terminologies*”

Definition (Deductive Difference)

$$\text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}') = \{\alpha \mid \alpha \text{ a } \mathcal{DL}\text{-axiom, } \mathcal{O} \not\models \alpha, \mathcal{O}' \models \alpha \text{ and } \text{Sig}(\alpha) \subseteq \Sigma\}$$

Definition (Deductive Difference for Mappings)

$$\text{mdiff}_{\Sigma_1, \Sigma_2}(\mathcal{O}, \mathcal{O}') = \{\alpha \in \text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}') \mid \alpha \text{ } \mathcal{DL}\text{-mapping between } \Sigma_1, \Sigma_2\}$$

Computation of New Entailments: *Semantic Difference*

How to extract new entailments in \mathcal{U}

- Notion of *deductive difference*
- B. Konev, D. Walther and Frank Wolter: “*The Logical Difference Problem for Description Logic Terminologies*”

Definition (Deductive Difference)

$$\text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}') = \{\alpha \mid \alpha \text{ a } \mathcal{DL}\text{-axiom, } \mathcal{O} \not\models \alpha, \mathcal{O}' \models \alpha \text{ and } \text{Sig}(\alpha) \subseteq \Sigma\}$$

Definition (Deductive Difference for Mappings)

$$\text{mdiff}_{\Sigma_1, \Sigma_2}(\mathcal{O}, \mathcal{O}') = \{\alpha \in \text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}') \mid \alpha \text{ } \mathcal{DL}\text{-mapping between } \Sigma_1, \Sigma_2\}$$

Computation of New Entailments: *Semantic Difference*

Problems of Semantic Difference...

- No algorithm for expressive DLs, such as *SROIQ* (OWL 2) and *SHOIQ* (OWL DL)
- Algorithms only for (fragments of) the OWL 2 EL and QL profiles
- The number of entailments in the difference can be huge (even infinite)

Necessity of an approximation

- $\text{diff}_{\Sigma}^{\approx}(\mathcal{O}, \mathcal{O}') \subseteq \text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}')$
- $\text{mdiff}_{\Sigma_1, \Sigma_2}^{\approx}(\mathcal{O}, \mathcal{O}') \subseteq \text{mdiff}_{\Sigma_1, \Sigma_2}(\mathcal{O}, \mathcal{O}')$

Computation of New Entailments: *Semantic Difference*

Problems of Semantic Difference...

- No algorithm for expressive DLs, such as *SROIQ* (OWL 2) and *SHOIQ* (OWL DL)
- Algorithms only for (fragments of) the OWL 2 EL and QL profiles
- The number of entailments in the difference can be huge (even infinite)

Necessity of an approximation

- $\text{diff}_{\Sigma}^{\approx}(\mathcal{O}, \mathcal{O}') \subseteq \text{diff}_{\Sigma}(\mathcal{O}, \mathcal{O}')$
- $\text{mdiff}_{\Sigma_1, \Sigma_2}^{\approx}(\mathcal{O}, \mathcal{O}') \subseteq \text{mdiff}_{\Sigma_1, \Sigma_2}(\mathcal{O}, \mathcal{O}')$

Approximations in ContentMap

A, B are atomic concepts (including \top, \perp) and R, S atomic roles

Approximation 1: reasoner output

- $A \sqsubseteq B$
- $R \sqsubseteq S$

Approximation 2: reasoner output plus ...

- $A \sqsubseteq \neg B$
- $A \sqsubseteq \exists R.B$
- $A \sqsubseteq \forall R.B$

Approximations in ContentMap

A, B are atomic concepts (including \top, \perp) and R, S atomic roles

Approximation 1: reasoner output

- $A \sqsubseteq B$
- $R \sqsubseteq S$

Approximation 2: reasoner output plus ...

- $A \sqsubseteq \neg B$
- $A \sqsubseteq \exists R.B$
- $A \sqsubseteq \forall R.B$

Outline

- 1 Introduction
- 2 Ontology Integration Method
- 3 Method Step 1: Mapping Computation
- 4 Method Step 2: Computation of Entailments
- 5 Method Step 3: Evaluation of Entailments**
- 6 Method Step 4: Repair of Unintended Entailments
- 7 Related Work
- 8 Evaluation and Conclusions

Evaluation of Entailments: ContentMap support

Manual Entailment selection

- Once we have

$$\Lambda = \text{diff}_{\Sigma_1}^{\approx}(\mathcal{O}_1, \mathcal{U}) \cup \text{diff}_{\Sigma_2}^{\approx}(\mathcal{O}_2, \mathcal{U}) \cup \text{mdiff}_{\Sigma_1, \Sigma_2}^{\approx}(\mathcal{M}, \mathcal{U})$$















- Some entailments may be intended (called \mathfrak{S}^+),
- while others may reveal potential errors in \mathcal{U} (called \mathfrak{S}^-).

Evaluation of Entailments: ContentMap support

Manual Entailment selection

- Once we have

$$\Lambda = \text{diff}_{\Sigma_1}^{\approx}(\mathcal{O}_1, \mathcal{U}) \cup \text{diff}_{\Sigma_2}^{\approx}(\mathcal{O}_2, \mathcal{U}) \cup \text{mdiff}_{\Sigma_1, \Sigma_2}^{\approx}(\mathcal{M}, \mathcal{U})$$
- Some entailments may be intended (called \mathfrak{S}^+),
- while others may reveal potential errors in \mathcal{U} (called \mathfrak{S}^-).

NEW ENTAILMENTS WRT SIGNATURE OF ONTOLOGY 1	NEW ENTAILMENTS WRT SIGNATURE OF ONTOLOGY 2
Negative_Factor ● Negative_Factor subClassOf Nothing (c: 0.4688)  	Negative_Rheum_Factor ● Negative_Rheum_Factor subClassOf Nothing (c: 0.4688)  
Positive_Factor ● Positive_Factor subClassOf Nothing (c: 0.4688)  	Positive_Rheum_Factor ● Positive_Rheum_Factor subClassOf Nothing (c: 0.4688)  
Oly_Juvenile_Arthritis ● Oly_Juvenile_Arthritis subClassOf Nothing (c: 0.4688)  	Rheum_Arthritis ● Rheum_Arthritis subClassOf Juv_Disease (c: 0.625)  
	Poly_Juv_Rheum_Arthritis ● Poly_Juv_Rheum_Arthritis subClassOf Nothing (c: 0.4688)  

Evaluation of Entailments: ContentMap support

Manual Entailment selection

- Once we have
$$\Lambda = \text{diff}_{\Sigma_1}^{\approx}(\mathcal{O}_1, \mathcal{U}) \cup \text{diff}_{\Sigma_2}^{\approx}(\mathcal{O}_2, \mathcal{U}) \cup \text{mdiff}_{\Sigma_1, \Sigma_2}^{\approx}(\mathcal{M}, \mathcal{U})$$
- Some entailments may be intended (called \mathfrak{S}^+),
- while others may reveal potential errors in \mathcal{U} (called \mathfrak{S}^-).

The screenshot displays two side-by-side windows from the ContentMap application, each showing a list of mapping entailments. The left window is titled "NEW MAPPING ENTAILMENTS (O1 → O2)" and the right window is titled "NEW MAPPING ENTAILMENTS (O2 → O1)".

Left Window (O1 → O2):

- Juvenile_Arthritis:**
 - JIA1:Juvenile_Arthritis subClassOf JIA2:Rheum_Arthritis (c: 0.8125) [I] [X] [C]
 - JIA1:Juvenile_Arthritis subClassOf JIA2:Juv_Rheum_Arthritis (c: 0.5) [I] [X] [C]
- Multi_Joint_Disease:**
 - JIA1:Multi_Joint_Disease subClassOf JIA2:Juv_Disease (c: 0.625) [I] [X] [C]
 - JIA1:Multi_Joint_Disease subClassOf JIA2:Disease (c: 1.0) [I] [X] [C]
- Rheumatoid_Arthritis:**
 - JIA1:Rheumatoid_Arthritis subClassOf JIA2:Juv_Disease (c: 0.625) [I] [X] [C]
 - JIA1:Rheumatoid_Arthritis subClassOf JIA2:Disease (c: 1.0) [I] [X] [C]

Right Window (O2 → O1):

- Rheum_Arthritis:**
 - JIA2:Rheum_Arthritis subClassOf JIA1:Disease (c: 1.0) [I] [X] [C]
- Juv_Rheum_Arthritis:**
 - JIA2:Juv_Rheum_Arthritis subClassOf JIA1:Disease (c: 1.0) [I] [X] [C]
- Systemic_Juv_Rheum_Arthritis:**
 - JIA2:Systemic_Juv_Rheum_Arthritis subClassOf JIA1:Rheumatoid_Arthritis (c: 0.8125) [I] [X] [C]

Evaluation of Entailments: ContentMap support

Dependency Relationship $\alpha \triangleright \beta$

- Entailments are organized within a dependency tree
- Based on the notion of Justification for an axiom
- $\text{Just}(\alpha, \mathcal{O})$ is the set of all justifications for α in \mathcal{O} .

Definition (Dependency Relationship)

Let $\mathcal{O} \models \alpha, \beta$. $\alpha \triangleright \beta$ iff for each $\mathcal{J}_\beta \in \text{Just}(\beta, \mathcal{O})$, there is $\mathcal{J}_\alpha \in \text{Just}(\alpha, \mathcal{O})$ s.t. $\mathcal{J}_\alpha \subseteq \mathcal{J}_\beta$.

Evaluation of Entailments: ContentMap support

Dependency Relationship $\alpha \triangleright \beta$

- Entailments are organized within a dependency tree
- Based on the notion of Justification for an axiom
- $\text{Just}(\alpha, \mathcal{O})$ is the set of all justifications for α in \mathcal{O} .

Definition (Dependency Relationship)

Let $\mathcal{O} \models \alpha, \beta$. $\alpha \triangleright \beta$ iff for each $\mathcal{J}_\beta \in \text{Just}(\beta, \mathcal{O})$, there is $\mathcal{J}_\alpha \in \text{Just}(\alpha, \mathcal{O})$ s.t. $\mathcal{J}_\alpha \subseteq \mathcal{J}_\beta$.

Dependency Relationship: ContentMap support

DEPENDENCY TREE OF NEW ENTAILEMENTS			
Entailments Dependency Tree			
	Keep	Remove	Conf... Justificat...
● JIA2:Juv_Rheum_Arthritis equivalentTo JIA2:Rheum_Arthritis	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.4063 ⓘ
└─● JIA2:Rheum_Arthritis subClassOf JIA2:Juv_Disease	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.4063 ⓘ
● JIA2:Positive_Rheum_Factor subClassOf not JIA1:Positive_Factor	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.625 ⓘ
└─● JIA1:Positive_Factor subClassOf not JIA2:Positive_Rheum_Factor	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.625 ⓘ
● JIA2:Positive_Rheum_Factor subClassOf Nothing	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.4688 ⓘ
└─● JIA2:Poly_Juv_Rheum_Arthritis subClassOf Nothing	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.4688 ⓘ
● JIA2:Negative_Rheum_Factor subClassOf not JIA1:Negative_Factor	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.625 ⓘ
└─● JIA1:Negative_Factor subClassOf not JIA2:Negative_Rheum_Factor	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.625 ⓘ
● JIA1:Rheumatoid_Arthritis subClassOf JIA2:Juv_Disease	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0.5 ⓘ
● JIA1:Juvenile_Arthritis subClassOf JIA2:Juv_Rheum_Arthritis	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.5 ⓘ
● JIA2:Positive_Rheum_Factor subClassOf not JIA1:Negative_Factor	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.75 ⓘ

Evaluation of Entailments: ContentMap support

Automatic Entailment selection

- Suggestions for \mathfrak{S}^+ and \mathfrak{S}^-
- **Unsatisfiable concepts** and entailments with **low confidence** will be included in \mathfrak{S}^-
- Entailments with **high confidence** will be included in \mathfrak{S}^+

Definition (Confidence of an Entailment)

- Confidence of a **mapping**: annotation value
- Confidence of an **axiom** α not annotated with a confidence value: $\text{conf}(\alpha) = 1$
- Confidence of a **Justification**: $\text{conf}(\mathcal{J}) = \prod_{\gamma \in \mathcal{J}} \text{conf}(\gamma)$
- Confidence of an **entailment** β :
$$\text{conf}(\beta) = \max(\bigcup_{\mathcal{J} \in \text{Just}(\beta, \mathcal{O})} \text{conf}(\mathcal{J}))$$

Evaluation of Entailments: ContentMap support

Automatic Entailment selection

- Suggestions for \mathfrak{S}^+ and \mathfrak{S}^-
- **Unsatisfiable concepts** and entailments with **low confidence** will be included in \mathfrak{S}^-
- Entailments with **high confidence** will be included in \mathfrak{S}^+

Definition (Confidence of an Entailment)

- Confidence of a **mapping**: annotation value
- Confidence of an **axiom** α not annotated with a confidence value: $\text{conf}(\alpha) = 1$
- Confidence of a **Justification**: $\text{conf}(\mathcal{J}) = \prod_{\gamma \in \mathcal{J}} \text{conf}(\gamma)$
- Confidence of an **entailment** β :
$$\text{conf}(\beta) = \max(\bigcup_{\mathcal{J} \in \text{Just}(\beta, \mathcal{O})} \text{conf}(\mathcal{J}))$$

Outline

- 1 Introduction
- 2 Ontology Integration Method
- 3 Method Step 1: Mapping Computation
- 4 Method Step 2: Computation of Entailments
- 5 Method Step 3: Evaluation of Entailments
- 6 Method Step 4: Repair of Unintended Entailments**
- 7 Related Work
- 8 Evaluation and Conclusions

Repair of Unintended Entailments

Extraction of Plans

- Given \mathfrak{S}^+ and \mathfrak{S}^-
- A repair plan for \mathcal{U} is a set of axioms $\mathcal{P} \subseteq \mathcal{U}$ such that:
 - $(\mathcal{U} \setminus \mathcal{P}) \models \alpha$ for each $\alpha \in \mathfrak{S}^+$, and
 - $(\mathcal{U} \setminus \mathcal{P}) \not\models \beta$ for each $\beta \in \mathfrak{S}^-$.

Note that conflicting choices in \mathfrak{S}^+ and \mathfrak{S}^- may make it impossible to find any plans.

Repair of Unintended Entailments

Extraction of Plans

- Given \mathfrak{S}^+ and \mathfrak{S}^-
- A repair plan for \mathcal{U} is a set of axioms $\mathcal{P} \subseteq \mathcal{U}$ such that:
 - $(\mathcal{U} \setminus \mathcal{P}) \models \alpha$ for each $\alpha \in \mathfrak{S}^+$, and
 - $(\mathcal{U} \setminus \mathcal{P}) \not\models \beta$ for each $\beta \in \mathfrak{S}^-$.

Note that conflicting choices in \mathfrak{S}^+ and \mathfrak{S}^- may make it impossible to find any plans.

Repair of Unintended Entailments: ContentMap Support

Organization of Candidate Plans

- ContentMap ranks the plans in order of:
 - number of affected axioms, or
 - confidence, $\text{conf}(\mathcal{P}) = \prod_{\alpha \in \mathcal{P}} \text{conf}(\alpha)$.

Extracted Plans

AVAILABLE REPAIR PLANS

Select an available plan from list

Plan 3 (confidence: 0.0477, axioms to remove: 6)

Axioms to Remove

Plan 3

JIA1:Disease subClassOf JIA2:Juv_Disease - (c: 0.625)	P	M
JIA1:Negative_Factor subClassOf JIA2:Positive_Rheum_Factor - (c: 0.625)		M
JIA2:Negative_Rheum_Factor subClassOf JIA1:Positive_Factor - (c: 0.625)		M
JIA2:Positive_Rheum_Factor subClassOf JIA1:Negative_Factor - (c: 0.625)		M
JIA1:Rheumatoid_Arthritis subClassOf JIA2:Juv_Rheum_Arthritis - (c: 0.5)	P	M
JIA1:Positive_Factor subClassOf JIA2:Negative_Rheum_Factor - (c: 0.625)		M

Use Plan Come Back

Outline

- 1 Introduction
- 2 Ontology Integration Method
- 3 Method Step 1: Mapping Computation
- 4 Method Step 2: Computation of Entailments
- 5 Method Step 3: Evaluation of Entailments
- 6 Method Step 4: Repair of Unintended Entailments
- 7 Related Work**
- 8 Evaluation and Conclusions

Related Work

Similar approaches

- **Debugging and revision of mappings:** Meilicke, C., Stuckenschmidt, H., Tamilin, A.
- **Debugging and repairing inconsistencies in OWL ontologies:** Kalyanpur et al. and Schlobach et al.

Our Contribution

- Our approach uses only **OWL 2 semantics**.
- We are not restricted to inconsistencies, but can include **any unintended entailment**.
- We provide techniques to evaluate entailments: **approximation of semantic difference, dependency tree**.
- We have **formalized the repair method**.
- We provide **tool support**.

Related Work

Similar approaches

- **Debugging and revision of mappings:** Meilicke, C., Stuckenschmidt, H., Tamilin, A.
- **Debugging and repairing inconsistencies in OWL ontologies:** Kalyanpur et al. and Schlobach et al.

Our Contribution

- Our approach uses only **OWL 2 semantics**.
- We are not restricted to inconsistencies, but can include **any unintended entailment**.
- We provide techniques to evaluate entailments: **approximation of semantic difference, dependency tree**.
- We have **formalized the repair method**.
- We provide **tool support**.

Outline

- 1 Introduction
- 2 Ontology Integration Method
- 3 Method Step 1: Mapping Computation
- 4 Method Step 2: Computation of Entailments
- 5 Method Step 3: Evaluation of Entailments
- 6 Method Step 4: Repair of Unintended Entailments
- 7 Related Work
- 8 Evaluation and Conclusions

Experiments

Bibliographic Ontologies: INRIA (\mathcal{O}_{INR}), MIT (\mathcal{O}_{MIT}), UMBC ($\mathcal{O}_{\text{UMBC}}$) and AIFB Karlsruhe ($\mathcal{O}_{\text{AIFB}}$).

Repair Using Gold Standard

- Unsatisfiability and unintended entailments were found

Synthetic Repair Using Mapping Tools

- We used OLA, AROMA and CIDER
- In all the cases ContentMap automatically found unsatisfiability and unintended entailments
- Repair plans corrected identified errors
- Precision (w.r.t. GS) was improved from 1-5%
- Recall decreased in some cases 1-3%

Experiments

Bibliographic Ontologies: INRIA (\mathcal{O}_{INR}), MIT (\mathcal{O}_{MIT}), UMBC ($\mathcal{O}_{\text{UMBC}}$) and AIFB Karlsruhe ($\mathcal{O}_{\text{AIFB}}$).

Repair Using Gold Standard

- Unsatisfiability and unintended entailments were found

Synthetic Repair Using Mapping Tools

- We used OLA, AROMA and CIDER
- In all the cases ContentMap automatically found unsatisfiability and unintended entailments
- Repair plans corrected identified errors
- Precision (w.r.t. GS) was improved from 1-5%
- Recall decreased in some cases 1-3%

Experiments

Bibliographic Ontologies: INRIA (\mathcal{O}_{INR}), MIT (\mathcal{O}_{MIT}), UMBC ($\mathcal{O}_{\text{UMBC}}$) and AIFB Karlsruhe ($\mathcal{O}_{\text{AIFB}}$).

Repair Using Gold Standard

- Unsatisfiability and unintended entailments were found

Synthetic Repair Using Mapping Tools

- We used OLA, AROMA and CIDER
- In all the cases ContentMap automatically found unsatisfiability and unintended entailments
- Repair plans corrected identified errors
- Precision (w.r.t. GS) was improved from 1-5%
- Recall decreased in some cases 1-3%

Conclusions

Some Drawbacks

- Problems of Scalability with big Ontologies
- The user is overwhelmed in some cases.

Possible Solution

Modularization and **Divide and Conquer** approach.

Conclusions

Some Drawbacks

- Problems of Scalability with big Ontologies
- The user is overwhelmed in some cases.

Possible Solution

Modularization and **Divide and Conquer** approach.

Feedback

We want you...

- ... to test **ContentMap** and give us feedback.



More information from ...

<http://krono.act.uji.es/people/Ernesto/contentmap>

Fancy a demo?

I'm happy to give one during any of the next breaks.

We want you...

- ... to test **ContentMap** and give us feedback.



More information from ...

<http://krono.act.uji.es/people/Ernesto/contentmap>

Fancy a demo?

I'm happy to give one during any of the next breaks.

Thank you!

We want you...

- ... to test **ContentMap** and give us feedback.



More information from ...

<http://krono.act.uji.es/people/Ernesto/contentmap>

Fancy a demo?

I'm happy to give one during any of the next breaks.

Questions?

We want you...

- ... to test **ContentMap** and give us feedback.



More information from ...

<http://krono.act.uji.es/people/Ernesto/contentmap>

Fancy a demo?

I'm happy to give one during any of the next breaks.

Thank you very much!