

推論結果へのヒューリスティックな手法の適用による OWL オントロジーデバッグ支援システムの試作

A Preliminary Approach for Ontology Debugging Support System Using Heuristic Approaches on Reasoning Results

山田 直希 *1
Yamada Naoki

福田 直樹 *2
Fukuta Naoki

*1静岡大学情報学部

Department of Computer Science, Shizuoka University

*2静岡大学大学院情報学領域

Department of Informatics, Shizuoka University

When an ontology is used by an inference engine, inconsistencies, a class corresponds nothing or other characteristic phenomenon appears on the result of reasoning. In this paper, we propose OWL ontology debug support system to find inconsistencies in an ontology and modify them by the features of reasoning results from concept hierarchy, and set relations in the ontology, as well as their query results when the ontology is used on a inference-enabled SPARQL endpoint.

1. はじめに

オントロジー記述言語 OWL[3] は記述論理 [4] に基づいたものであり、記述論理に基づく推論が可能である。OWL で用いられる推論にも適用可能な推論器の実装として、HermiT[5] や Pellet[6] などが提案されている。推論のトレースを取ることで自体に特別な手法が必要となる場合がある [7]。

エンドポイントにおいて、推論付きクエリを実行する場合、推論をおこなわずにクエリを実行する場合に比べて時間がかかり、推論付きクエリの実行性能の予測としては文献 [2] がある。文献 [2] では、特殊な中継サーバを用いて推論負荷の大きいクエリの負荷を軽減する仕組みが提案されている。推論の負荷を軽減するためにクエリではなくオントロジーを改変することに着目し、オントロジーの改変がエンドポイントでの推論に与える影響を検証するシステムの試作が文献 [1] でされている。

本研究ではオントロジーが推論結果において矛盾を含む場合、その矛盾の原因を検出する方法と原因に対する修正を提案する手法に基づいて、オントロジーの修正を支援するシステムの設計について述べる。これまでに、オントロジーのデバッグについて提案しているものとして文献 [8] や文献 [9] がある。文献 [8] では、提案した手法をオントロジーエディタ Swoop で実装している。本研究で述べるオントロジー修正支援システムでは、オントロジーの推論結果のみによるアプローチと異なり、文献 [1] で試作したシステムを利用することでオントロジーを推論付きエンドポイントに利用した場合のクエリ実行結果の変化も修正の指針として用いることができる。

2. 推論結果による矛盾の検出と矛盾に対する修正の例

2.1 推論結果における矛盾に基づく検出

オントロジー内での定義内容の論理的な不整合については、そのオントロジーに対する TBox 推論 [3] を行った際の矛盾の検出によって確認することができる場合がある。例えば、pizza オントロジーにおいて、推論をおこなうと CheeseyVegetableTopping と IceCream の 2 つのクラスが Nothing クラスと同値関係になる。CheeseyVegetableTopping と IceCream の記述はそれぞれ Listing1, Listing2 となっているが、推論結果における CheeseyVegetableTopping の記述は Listing3, IceCream の記述は Listing4 となる。

Listing1 からわかるように、CheeseyVegetableTopping の親クラスは CheeseTopping と VegetableTopping である。CheeseTopping と VegetableTopping の関係は Listing5 のようになっており、CheeseyVegetableTopping の 2 つの親クラスが Disjoint With な関係であることから、推論結果に矛盾が生じることになる。IceCream の親クラスは Food と hasTopping some FruitTopping で、hasTopping は Listing7 のようになっており、hasTopping の Domain は Pizza だが、IceCream は Pizza の子クラスではない。また、Listing6 からわかるように、hasTopping の Domain になっている Pizza と IceCream は Disjoint With な関係なので推論結果に矛盾が生じている。

```
<owl:Class rdf:about="#CheeseyVegetableTopping">
  <rdfs:label xml:lang="pt">
    CoberturaDeQueijoComVegetais</rdfs:label>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#CheeseTopping" />
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#VegetableTopping" />
  </rdfs:subClassOf>
</owl:Class>
```

Listing 1: CheeseyVegetableTopping の記述

```
<owl:Class rdf:about="#IceCream">
  <rdfs:label xml:lang="pt">Sorvete</rdfs:label>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasTopping"/>
      <owl:someValuesFrom rdf:resource="#FruitTopping"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Food" />
  </rdfs:subClassOf>
</owl:Class>
```

Listing 2: IceCream の記述

```
<owl:Class rdf:about="http://www.co-ode.org/ontologies/pizza/pizza.owl#CheeseyVegetableTopping">
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Nothing" />
```

連絡先: 山田 直希, 静岡大学情報学部情報科学科,
〒 432-8011 静岡県浜松市中区城北 3-5-1, E-mail:
cs13098@s.inf.shizuoka.ac.jp

```

<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#ChickenTopping"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#Country"/>
.
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#Veneziana"/>
<owl:disjointWith rdf:resource="http://
www.w3.org/2002/07/owl#Nothing"/>
<owl:disjointWith rdf:resource="http://
www.w3.org/2002/07/owl#Thing"/>
</owl:Class>

```

Listing 3: 推論結果における CheeseyVegetableTopping の記述

```

<owl:Class rdf:about="http://www.co-ode.org/
ontologies/pizza/pizza.owl#IceCream">
<rdfs:subClassOf rdf:resource="http://www
.w3.org/2002/07/owl#Nothing"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#InterestingPizza"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#JalapenoPepperTopping"/>
.
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#Veneziana"/>
<owl:disjointWith rdf:resource="http://
www.w3.org/2002/07/owl#Nothing"/>
<owl:disjointWith rdf:resource="http://
www.w3.org/2002/07/owl#Thing"/>
</owl:Class>

```

Listing 4: 推論結果における IceCream の記述

```

<owl:Class rdf:about="#CheeseTopping">
<owl:disjointWith>
<owl:Class rdf:about="#
VegetableTopping"/>
</owl:disjointWith>
</owl:Class>
<owl:Class rdf:about="#VegetableTopping">
<owl:disjointWith>
<owl:Class rdf:about="#CheeseTopping"
/>
</owl:disjointWith>
</owl:Class>

```

Listing 5: CheeseTopping と VegetableTopping の関係

```

<owl:Class rdf:about="#Pizza">
<owl:disjointWith>
<owl:Class rdf:about="#IceCream"/>
</owl:disjointWith>
</owl:Class>
<owl:Class rdf:about="#IceCream">
<owl:disjointWith>
<owl:Class rdf:about="#Pizza"/>
</owl:disjointWith>
</owl:Class>

```

Listing 6: Pizza と IceCream の関係

```

<owl:ObjectProperty rdf:about="#hasTopping">
<rdfs:type rdf:resource="&owl;
InverseFunctionalProperty"/>
<owl:inverseOf>
<owl:ObjectProperty rdf:about="#
isToppingOf"/>
</owl:inverseOf>
<rdfs:domain>
<owl:Class rdf:about="#Pizza"/>
</rdfs:domain>
<rdfs:range>
<owl:Class rdf:about="#PizzaTopping"/>
</rdfs:range>
</owl:ObjectProperty>

```

Listing 7: hasTopping の記述

2.2 推論結果における矛盾の修正

2.1 節で示した例では、IceCream の親クラスの記述で用いられているロールの hasTopping は IceCream とは Disjoint With な関係の Pizza を Domain としており、それが推論結果において矛盾が生じる原因と考えられることから、その矛盾の解消方法として単純な方法としては、Listing8 のような Domain が IceCream の新しいロールを作成し、hasTopping と入れ替えることが考えられる。この変更を適用した場合の推論結果における IceCream の記述は Listing9 のようになり、先の矛盾については解消できていることが確認できる。また、ロールは変更せずに IceCream と Pizza の Disjoint 関係を削除した場合も推論結果における矛盾は解消できるが、推論結果における IceCream の記述は Listing10 のようになり、新しいロールを追加した場合とは異なり、推論結果において IceCream が Pizza の subclasses になる。このように、矛盾が起きないようにオントロジーを修正する方法には複数通りのものがあると考えられる場合があり、それらのうちのどの修正方法をとるべきかが、容易に決定できない場合がある。

3.2 節で CheeseyVegetableTopping が推論結果において矛盾を生じる原因は、親クラスになっている CheeseTopping と VegetableTopping がお互いに Disjoint With な関係にある点に起因している。この矛盾をもっとも簡潔に解消するには、CheeseTopping と VegetableTopping の DisjointWith 関係の記述を削除することであり、この場合の推論結果における CheeseyVegetableTopping の記述は Listing11 のとおりであり、矛盾を解消できていることが確認できる。このような改変は、推論時の矛盾を解消できる一方で、そこで記述可能とすべき概念の表現そのものも諦めてしまっていることから、その妥当性について、何らかの方法での検証が必要になると考えられる。

```

<owl:ObjectProperty rdf:about="http://www.co-
ode.org/ontologies/pizza/pizza.owl#
OWLObjectProperty_80e7908f-d48a-4c55-
9784-f331bba48f19">
<rdfs:label xml:lang="ja">
hasTopping_IceCream</rdfs:label>
<rdfs:domain rdf:resource="http://www.co-
ode.org/ontologies/pizza/pizza.owl#
IceCream"/>
<rdfs:range rdf:resource="http://www.co-
ode.org/ontologies/pizza/pizza.owl#
PizzaTopping"/>
<rdfs:subPropertyOf rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#hasIngredient"/>
</owl:ObjectProperty>

```

Listing 8: hasTopping の代わりに追加したロール

```

<owl:Class rdf:about="http://www.co-ode.org/
ontologies/pizza/pizza.owl#IceCream">
<rdfs:subClassOf rdf:resource="http://www
.co-ode.org/ontologies/pizza/pizza
.owl#Food"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#InterestingPizza"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#JalapenoPepperTopping"/>
.
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#VegetarianTopping"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#Veneziana"/>

```

```

<owl:disjointWith rdf:resource="http://
www.w3.org/2002/07/owl#Nothing"/>
</owl:Class>

```

Listing 9: ロールを追加した場合の推論結果における IceCream の記述

```

<owl:Class rdf:about="http://www.co-ode.org/
ontologies/pizza/pizza.owl#IceCream">
<rdfs:subClassOf rdf:resource="http://www
.co-ode.org/ontologies/pizza/pizza
.owl#Pizza"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#JalapenoPepperTopping"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#LaReine"/>
.
.
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#VegetableTopping"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#VegetarianTopping"/>
<owl:disjointWith rdf:resource="http://
www.w3.org/2002/07/owl#Nothing"/>
</owl:Class>

```

Listing 10: Pizza と IceCream の Disjoint With 関係を削除した場合の推論結果における IceCream の記述

```

<owl:Class rdf:about="http://www.co-ode.org/
ontologies/pizza/pizza.owl#
CheesyVegetableTopping">
<rdfs:subClassOf rdf:resource="http://www
.co-ode.org/ontologies/pizza/pizza
.owl#CheeseTopping"/>
<rdfs:subClassOf rdf:resource="http://www
.co-ode.org/ontologies/pizza/pizza
.owl#VegetableTopping"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#ChickenTopping"/>
.
.
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#VegetarianPizzaEquivalent2"/>
<owl:disjointWith rdf:resource="http://
www.co-ode.org/ontologies/pizza/pizza
.owl#Veneziana"/>
<owl:disjointWith rdf:resource="http://
www.w3.org/2002/07/owl#Nothing"/>
</owl:Class>

```

Listing 11: CheeseTopping と VegetableTopping の Disjoint With 関係を削除した場合の推論結果における CheesyVegetableTopping の記述

3. オントロジー修正支援システムの設計

3.1 システムの構成

本研究で試作を進めているシステムの構成を図 1 に示す。デバッグ支援システムには OWL API と推論器の HermiT が組み込まれており、OWL API によって入力されたオントロジーの解析や、書き出しを行い、HermiT で推論をおこなう。HermiT で推論した結果に矛盾が見られる場合は、その推論結果を用いて矛盾の原因と推測されるものを検出する。ユーザに対して、矛盾の原因として推測されるものと、それを修正する方法を提示し、ユーザが選択した修正方法によってオントロジーの修正をおこなう。修正したオントロジーに対しても推論をおこない、その修正がオントロジーの推論結果に対してどのように影響したかを確認する。

また、外部のプログラムによって指定したオントロジーを用いたエンドポイントを構築し、そのエンドポイントに対して複

数のクエリを実行することができ、修正前のオントロジーと修正後のオントロジーをインデックスとして用いた推論付きエンドポイントにおいてのクエリ実行結果の変化も修正の指針として用いることができる。

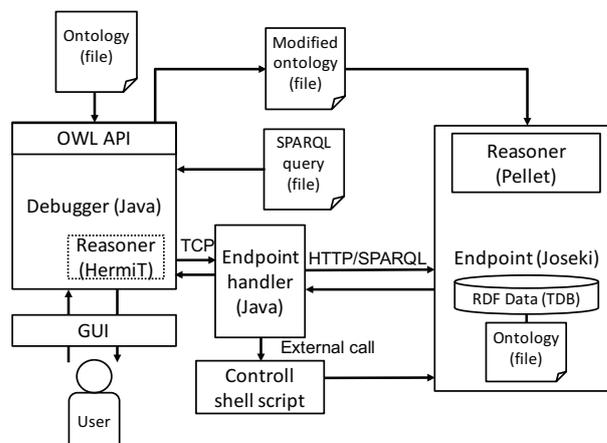


図 1: システム構成図

システムのユーザインタフェースは GUI で与えられており、読み込んだオントロジーのクラスとプロパティの階層、個々のクラスやプロパティについての記述、推論結果から推測した矛盾の原因とその修正方法、エンドポイントに対して実行するクエリとその結果をそれぞれ GUI で表示する。

システムのユーザインタフェースを図 2 に示す。

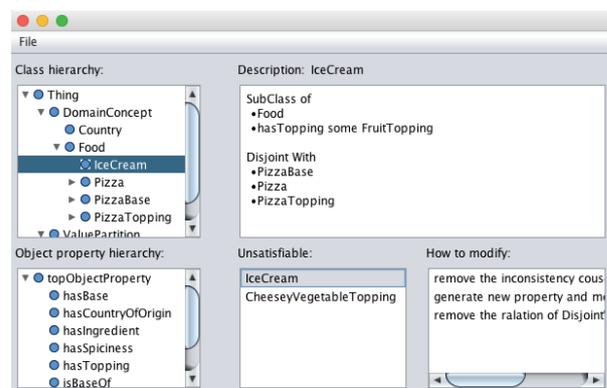


図 2: ユーザインタフェース図

3.2 矛盾の原因の検出とその修正支援

まずはじめに対象となるオントロジーの推論をおこなう。そのオントロジーの推論結果に矛盾が含まれる場合は、充足不能になったクラスを 1 つずつ参照し、矛盾の原因と推測できる部分を探す。

矛盾の原因として考えられるものの例を以下に示す。

- クラスの記述で直接の親クラスになっているクラスや、親クラスの記述で用いられるプロパティの Range になるクラスなど、そのクラスと関連しているクラスが充足不能になっている。
- 複数のクラスが親クラスとして記述されていて、その親クラス同士が Disjoint With な関係にある。

- 親クラスとして記述されているクラスと Disjoint With な関係にある。
- 親クラスの記述にプロパティが用いられていて、クラスの中でプロパティの Range として記述されているクラスがそのプロパティの定義で Range になっているクラスと Disjoint With な関係として定義されている。

本研究で試作しているシステムでは、充足不能なクラスに対して以上に示したような特徴がないか調べることで、矛盾の原因となっている部分を推測する。

2.1 節で示した矛盾の原因と考えられるものを示したものが、上記の方法で検出できることを示す。

CheesyVegetableTopping には 2 つの親クラスが存在し、それらが Disjoint With な関係として定義されているので、その部分が矛盾の原因として推測される。

IceCream は親クラスの記述にプロパティの hasTopping が用いられており、そのプロパティの Domain となるクラスが IceCream と Disjoint With な関係になっているので、親クラスになっているクラスと Disjoint With な関係になっているとして、その部分が矛盾していると推測できる。

3.3 矛盾の修正方法

3.2 節で示したそれぞれの矛盾の原因と推測できるものに対する修正方法を示す。

関連するクラスが充足不能な場合は関連するクラスを修正し、そのクラス自身には手を加えない。または、充足不能なクラスとの関連を記述している部分自体を削除する。

複数の親クラス同士が Disjoint With な関係にある場合は、その親クラス同士が Disjoint With となっている記述を削除するか、Disjoint With の関係にあるクラス同士が親クラスにならないように親クラスの記述を削除する。

親クラスの記述に用いられているプロパティに問題がある場合、例えば、親クラスの記述内で Range とされているクラスがプロパティの Range として定義されているクラスの子クラスか同値関係にない場合は、条件に合うプロパティを新たに生成し、そのプロパティを代わりに用いるようにするか、そのプロパティを用いた親クラスの記述自体を削除する。

3.4 エンドポイントにおけるクエリ実行結果の利用

本研究で試作しているシステムには、任意のオントロジーを利用した推論付きエンドポイントを起動し、そのエンドポイントに対してクエリを実行することができるシステムが組み込まれている。本システムを利用することで修正前のオントロジーを用いたエンドポイントへのクエリ実行結果と修正後のオントロジーを用いたエンドポイントへのクエリ実行結果を比較することができる。

このように、本試作システムでは、オントロジーを修正することで推論結果がどのように変化するかということに加え、オントロジーを推論付きエンドポイントで利用した場合に、オントロジーの修正によってクエリ実行結果がどう変化するかということを修正の指針として利用することができる。

4. おわりに

推論エンジンが推論の対象とするオントロジーにおいて、そのオントロジー内に矛盾が含まれている場合、該当するクラスが存在しないものとして扱われる等、特徴的な結果が推論結果に現れる場合がある。本研究では、推論エンジンによる推論結果の特徴とオントロジー内の概念階層や概念の集合関係に注

目して、オントロジーの矛盾の原因になっていると推測される記述を検出し、その検出結果に対する修正方法を提案、修正するオントロジーデバッグ支援システムの試作について述べた。今後の課題としては、実際のオントロジーの作成過程 [10] を考慮した支援の実現が考えられる。

5. 謝辞

本研究の一部は、JST CREST の支援を受けたものである。

参考文献

- [1] 山田 直希, 山形 祐史, 福田 直樹: 推論を考慮した LOD エンドポイントの構築支援機構のためのオントロジー修正支援機構の試作. 人工知能学会セマンティックウェブとオントロジー (SWO) 研究会 (2016)
- [2] Yamagata, Y., Fukuta, N.: An Approach to Dynamic Query Classification and Approximation on an Inference-enabled SPARQL End-point. *Journal of Information Processing*. Vol. 23, No. 6, pp. 759-766 (2015)
- [3] Antoniou, G., Van Harmelen, F.: *Web Ontology Language: OWL*. In: *Handbook on ontologies*, pp. 91-110. Springer-Verlag (2009)
- [4] Baader, F., Horrocks, I., Sattler, U.: *Description Logics*. In: *Handbook on Ontologies*, pp. 21-43. Springer-Verlag (2009)
- [5] Motik, B., Shearer, R., Horrocks, I.: *Hypertableau Reasoning for Description Logics*. *Journal of Artificial Intelligence Research* 36, 165-228 (2009)
- [6] Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: *Pellet: A practical owl-dl reasoner*. *Journal of Web Semantics* 5(2), 51-53 (2007)
- [7] Kazakov, Y., Klinov, P.: *Goal-Directed Tracing of Inferences in EL Ontologies*. In: Mika, P., Tudorache, T., Bernstein, A., Welty, C., Knoblock, C., Vrandečić, D., Groth, P., Noy, N., Janowicz, K., Goble, C. (Eds.) *The Semantic Web-ISWC 2014 Part II*, pp. 196-211. Springer-Verlag (2014)
- [8] Parsia, B., Sirin, E., Kalyanpur, A.: *Debugging OWL Ontologies*. *Proceedings of the 14th international conference on World Wide Web (WWW2005)*, pp. 633-640 (2005)
- [9] Wang, H., Horridge, M., Rector, A.: *Debugging OWL-DL Ontologies: A Heuristic Approach*. Y. Gil et al. (Eds.): *ISWC 2005, LNCS 3729*, pp. 745-757, 2005. Springer-Verlag Berlin Heidelberg (2005)
- [10] Walk, S., Singer, P., Noboa, L. E., Tudorache, T., Musen, M. A., Strohmaier, M.: *Understanding How Users Edit Ontologies: Comparing Hypotheses About Four Real-World Projects*. In: M. Arenas et al. (Eds.): *ISWC 2015, Part I, LNCS 9366*, pp. 551-568. Springer-Verlag (2015)