Wikipedia オントロジーに基づく ドメインオントロジー構築支援環境の実現と評価

A Domain Ontology Development Environment with Wikipedia Ontology

桜井 慎弥[*1] 手島拓也[*1] 森田 武史[*1] 和泉 憲明[*2] 山口 高平[*1]
Shinya Sakurai Takuya Tejima Takeshi Morita Noriaki Izumi Takahira Yamaguchi

[*1] 慶應義塾大学 Keio University
[*2] 独立行政法人 産業技術総合研究所 National Institute of Advanced Industrial Science and Technology

In this paper, we propose a way to extract effective relationships between concepts from the resources of Wikipedia. We applied several techniques on Wikipedia Japan, and have extracted five types of data, word linkages, synonymous words, class-instance relationships, is-a relationships and property information. We call it Wikipedia Ontology. In addition, we implemented a domain ontology development environment that uses information resources of Wikipedia Ontology. We evaluated how our approach supports concept definition in domain ontologies.

1. はじめに

ドメインオントロジーは専門分野に特化した情報検索やデータ統合において有用であるが，人手による構築コストは大きい．現在，フリーテキストとは異なるハイパーリンクやフィードを活用した半構造化情報資源が広がりをみせている．中でも情報鮮度・語彙網羅性の優れた百科事典 Wikipedia がその代表例であり，オントロジー構築のためのリソースとして注目を集めている．ただし，それらの情報資源はユーザ参加型という性質上，厳密な体系化が行われていないため，オントロジーへ直接結び付けることは難しい．そこで我々はこれまで，Wikipedia から大規模なオントロジー（Wikipedia オントロジー）を構築する手法の提案をしてきた． Wikipedia オントロジーについては，3 章で詳述する．

本研究では， Wikipedia カテゴリツリーと Infobox テンプレートとのマッチング処理を行うことで，これまで定義できなかった Is-a 関係とプロパティ定義域を抽出し，Wikipedia オントロジーの情報量を増加させる．また，Wikipedia オントロジーを利用したドメインオントロジー構築支援ツール「TDO」の実装とその評価を行う．

2. 関連研究

本章は，Wikipedia からオントロジーを構築する主な研究を紹介する．

DBPedia[Auer 07]は，Wikipedia の半構造化情報を RDF に変換することによって，大規模なデータベースを構築した．リソースとしては主に，英語 Wikipedia の Infobox や外部リンク，所属カテゴリといった半構造情報を利用している．

YAGO[Fabian 07]は，*Conceptual Category* と呼ばれるカテゴリをクラスとして利用し，WordNet を拡張している．*Conceptual Category* とは英語 Wikipedia のカテゴリであり，“American singers of German origin”カテゴリのように，カテゴリ名の *head* 部分である“singers”が複数形になっているカテゴリのことである．インスタンスに関しては，*BORNINYEAR* や *LOCATEDIN* といった *Relation* を用いてメタデータを記述し，非階層構造も構築している．

連絡先：桜井慎弥，山口高平，慶應義塾大学理工学部
〒223-8522 神奈川県横浜市港北区日吉 3-14-1
Tel: 045-566-1614，E-mail: {s_saku,yamaguti}@ae.keio.ac.jp

3. Wikipedia オントロジー

本研究では，Wikipedia を利用して構築したオントロジーを Wikipedia オントロジーと呼ぶ．

[手島 07]では Wikipedia のマイニングによる語彙の関連度定義とリダイレクトリンクを利用した同義語定義を行い，[桜井 08]ではカテゴリツリーに対するマッチング処理による Is-a 関係の定義と一覧記事に対するスクレイピングによるインスタンスの収集を行った．

3.1 Wikipedia オントロジーの構成

以下に示すように，Wikipedia オントロジーは六つの関係定義を行うことによって構築される．本研究では，4 の Is-a 関係，6 のプロパティ定義域の定義手法として Infobox テンプレート名とカテゴリ名のマッチングを提案する．

1. 関連関係(skos:related)
2. シノニム(skos:prefLabel, skos:altLabel)
3. クラス-インスタンス関係（rdf:type）
4. Is-a 関係(rdfs:subClassOf)
5. Infobox トリプル
6. プロパティ定義域（rdfs:domain）

3.2 Infobox テンプレート名とカテゴリ名の照合による Is-a 関係とプロパティ定義域の抽出

(1) Is-a 関係の抽出

Infobox とは，テーブルを利用して Wikipedia の記事の属性(Wikipedia では主に“項目”と呼ばれている)と属性値を整理して表示しているもので，記事の中にしばしば掲載されている．ここで使用される項目が，ドメインごとにある程度フォーマット化されているということが大きな特徴である．例えば「Java」の記事に掲載されている Infobox には“開発者”や“プラットフォーム”などの項目とそれぞれに対応する値が記述されており，この“開発者”や“プラットフォーム”という項目は，Infobox のテンプレート“プログラミング言語”で定められている．本研究は，各 Infobox の持つ抽象的なテンプレート名と，領域によっては多くの具体

的な概念を持つカテゴリ名との関係に着目する．テンプレート名とカテゴリ名の照合を行い Is-a 関係を抽出する．抽出の手順を以下の 1～4 に示す．また，具体的な手法を表しているのが図 1 である．

1. カテゴリとテンプレートの情報をデータベースに格納
2. カテゴリ名とテンプレート名の単純文字列照合
3. 照合したカテゴリ以下に存在するサブカテゴリ名と，照合したテンプレートを持つ記事が所属する全てのカテゴリ名とのマッチング
4. マッチングによって得られたサブカテゴリ名をテンプレート名と Is-a 関係が成り立つとして抽出

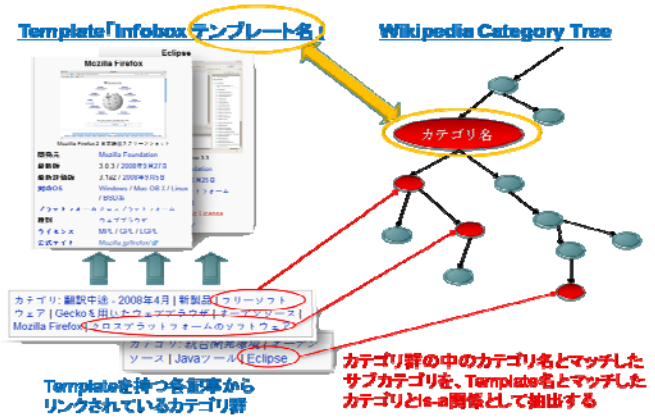

図 1 Infobox テンプレートとカテゴリ名の照合

以上の手順を実行することによって，[桜井 08]で行った文字列の特性を利用した「カテゴリ階層に対する文字列マッチング」では抽出することのできなかった Is-a 関係を抽出できる．それに伴い，正しくない Is-a 関係を多く持つ Wikipedia カテゴリツリーの洗練が可能になると考えられる．また，3.2 (2)において，ここで抽出した Is-a 関係と Infobox の持つプロパティとの関係に着目することで，プロパティ定義域の抽出を行う．

(2) プロパティ定義域の抽出

Infobox を有する記事－項目－値という三つ組は，RDF トリプルと捉えることができ主語である記事が属するカテゴリを調べることで，プロパティの定義域を定義できる可能性を持つ．図 2 は，Infobox と RDF トリプルとの対応関係と，記事「Ruby」が属するカテゴリが「設計者」というプロパティの定義域として定義される可能性を持つことを表している．

図 2 Infobox と RDF トリプル，カテゴリと定義域の対応

本研究では以下に述べる手法でプロパティ定義域の抽出を試みる．まず，テンプレート名を，Infobox が持つ各プロパティの定義域として抽出する．次に，3.2(1)で得られた Is-a 関係として正しいサブカテゴリを，テンプレートの持つ各プロパティの定義域として対応付ける．さらに，テンプレートで定義されていないプロパティの定義域抽出を試みる．

実際に記事で使用されている Infobox に登場するプロパティは，テンプレートで定義されているプロパティ以外のものが使用されるケースが多数存在する．例えば，「有機化合物」というテンプレートで定義されているプロパティは「構造式」，「形状」，「沸点」など合計 21 あるが，実際の記事に掲載されている Infobox のソースから収集したプロパティは，「CAS 登録番号」，「揮発性」，「臭気」，「蒸気圧」などテンプレートで定義されていないものが多く存在し合計 33 のプロパティを持つ．

そこで，本提案手法では，3.2(1)得られたサブカテゴリと，そのカテゴリに属する記事が持つ Infobox テンプレートで定義されていないプロパティとの関係に着目する．そうすることで，各プロパティの定義域として最上位の概念であるテンプレート名が得られるだけではなく，より具体化され，ドメインに特化したプロパティおよび定義域の抽出が可能となる．図 3 がプロパティ定義域の抽出手法の全体像である．

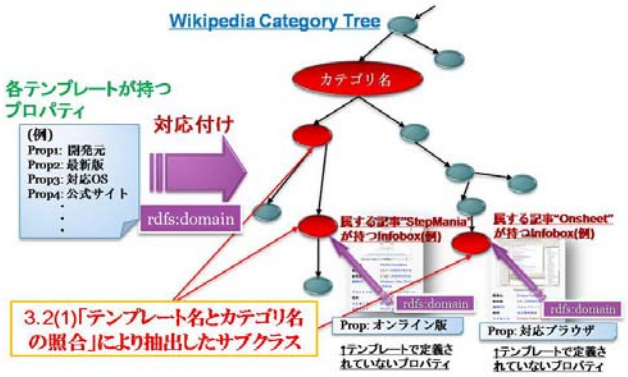

図 3 テンプレートで定義されていないプロパティ定義域の抽出

4. ドメインオントロジー構築支援ツール TextToDomainOntology

3 章では，Wikipedia オントロジーの構築手法について述べた．本節では，我々が開発した Wikipedia オントロジーを利用したドメインオントロジー構築支援ツールである TDO(TextToDomainOntology)について述べる．このツールは，ある専門文書をインプットすると，Wikipedia オントロジーを参照してドメインオントロジーとしてアウトプットするツールである．図 4 に TDO のシステムアーキテクチャを示す．

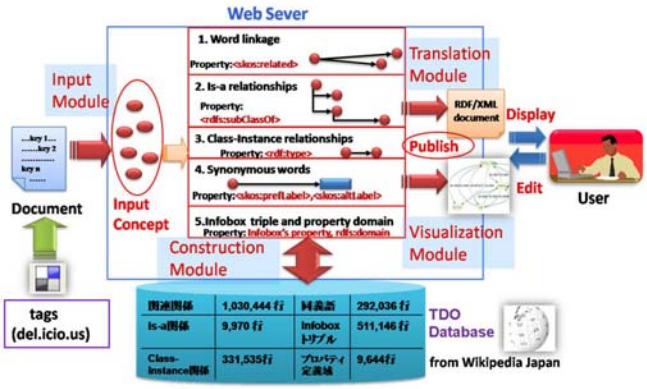

図 4 TDO のシステムアーキテクチャ

表 1 に示す情報がデータベースに格納されており，インプットされたドキュメントデータに応じて必要なデータを返す構造とな

っている. また, 出力は RDF/XML 形式をサポートしている. ウェブブラウザ上で手軽かつ容易にオントロジー構築が可能となる環境を目指す.

表 1 TDO の DB に格納されている関係の種類と行数

関連関係と関連度(skos:related)	同義語(skos:prefLabel, altLabel)	クラスーインスタンス関係(rdf:type)
1,030,444	292,036	331,535
Is-a 関係 (rdfs:subClassOf)	Infobox から得られるトリプル	プロパティ定義域 (rdfs:domain)
9,970	511,146	9,644

TDO は Input モジュール, Construction モジュール, Translation モジュール, Visualization モジュールという 4 つのモジュールが動作し, ドメインオントロジー構築支援を実現する.

Input モジュールは,入力されたドキュメントから, Wikipedia の記事名と Infobox の持つプロパティ名とマッチするキーワードを抽出する. また, 抽出した語彙をユーザに提示し, 選択された語彙を Construction モジュールに送る. Construction モジュールは Input モジュールにより入力された概念を表 1 で示したデータの中から検索し,自動的に概念間の関連を構築する機能である. Construction モジュールにより構築された概念間の関係から RDF/XML のデータを生成する機能が Translation モジュール, 構築された関係をグラフとしてユーザに提供する機能が Visualization モジュールである.

以下に, TDO の提供する特徴的な機能をまとめる.

- オントロジー半自動構築機能
- 入力語から RDF ソース・グラフ生成機能
- Table view による RDF 編集機能
- Source view, XML Outline 機能
- 関連概念検索機能
- del.icio.us タグのインポート機能
- 既存 RDF のインポート機能

5. Wikipedia オントロジーの評価実験

5.1 実験方法

2008 年 5 月の Wikipedia のダンプデータをダウンロードして, 実験を行った. データベースは MySQL, 実装言語は Java を使用した.

5.2 Infobox テンプレート名とカテゴリ名の照合による Is-a 関係とプロパティ定義域の抽出結果

(1) Is-a 関係の抽出結果

XML 形式で提供されている日本語 Wikipedia 全記事ソースデータ(pages-articles.xml.bz2)から 50,620 の Infobox を抽出した. Infobox が持つテンプレートを調べた結果, 717 種類存在した. 3.2(1)で述べた手法を用いた結果を表 2 にまとめる.

また, 提案手法によりマッチしたサブカテゴリ数でソートした結果の上位を表 3 に示す.

表 2 Infobox テンプレート名とカテゴリ名の照合結果

Wikipedia カテゴリの数	Infobox テンプレートの種類	テンプレート名・カテゴリ名の照合数	Is-a 関係として抽出されたサブカテゴリ数
51,427	717	195	2,778

表 3 提案手法によりマッチしたカテゴリの持つサブカテゴリ数

カテゴリ名(テンプレート名)	提案手法によりマッチしたサブカテゴリ数
企業	1436
国	302
テレビ番組	237
ソフトウェア	142
アナウンサー	105
山	84

(2) プロパティ定義域の抽出結果

テンプレート名を, Infobox が持つ各プロパティの定義域として抽出を行った. 存在する全 711 のテンプレートに対して Infobox の抽出を行い, プロパティの合計を求めたところ, 9,644 であった. つまり, 9,644 のプロパティそれぞれがテンプレート名を定義域として持つ結果が得られた.

次に, 3.2(1)で得られたサブカテゴリを定義域として対応付けを行った. 通常最上位の概念を定義域としていれば Is-a 関係が成り立つ下位概念を定義域としてみなす必要はないが, この対応付けはドメインオントロジー構築支援としてどれだけ多くのプロパティ定義域を自動的に生成することができるのかを示す指標となる. 結果的に, 合計 95,134 の対応付けが得られた.

さらに, テンプレートで定義されていないプロパティの定義域抽出をいくつかのテンプレートを対象に行った. なお, 3.2(1)で述べた手法で得られたテンプレートにおいて, テンプレートが持つプロパティ種類数と, 記事から抽出したプロパティ種類数の比較を行った. その結果, テンプレートが持つプロパティ種類数の平均は 18.9, 記事から抽出したプロパティ種類数の平均は 36.2 であった. 記事から抽出した Infobox が持つプロパティ種類数はテンプレートで定義されているプロパティの約 2 倍であることがわかる.

5.3 Infobox テンプレート名とカテゴリ名の照合による Is-a 関係とプロパティ定義域の抽出結果の考察

表 2 に示すように, テンプレート名とカテゴリ名がマッチしたカテゴリ(以下, 本稿ではルートカテゴリと呼ぶ)は 195 存在した. しかし, 各ルートカテゴリにおいて Infobox を持つ記事が属するカテゴリ群とマッチしたサブカテゴリを持つものは 43 であった. その理由は, ルートカテゴリの中に「オリンピック・・選手団」という, サブカテゴリを一つも持たないカテゴリが 114 存在したからである. 表 4 が, 3.2(1)で述べた手法によって抽出できた 43 の各ルートカテゴリに対して, Is-a 関係の評価を行った結果の一部である. なお, Is-a 関係が正しく成り立つと人手によって判断された各ルートカテゴリ以下のサブカテゴリを正解集合として再現率を算出している.

提案手法により抽出された Is-a 関係の正答率は 92%であり非常に精度が高く抽出ができたといえる. そして,「楽器」を例に挙げると, [桜井 08]で提案した「カテゴリ階層の複合語に対する照合」の手法では得ることのできない「ピアノ」や「トランペット」といった下位概念が抽出できている.

再現率に関しては 55.7%という結果が得られた.「有機化合物」のように, 抽出した Is-a 関係が正解集合と完全一致したケースもあるが,「銀行」のように, 正解の Is-a 関係がサブカテゴリ以下に 35 存在しているにも関わらず, 抽出された Is-a 関係は 3 であったケースもあった. 全体的に再現率が低くなった理由はいくつか考えられるが, 我々は, カテゴリツリーの大きさに対して Infobox を持つ記事の量が非常に小さいという理由が主であると考えている. カテゴリツリーは正しい Is-a 関係を多数含むもの

の, 性質の継承という観点から捉えた際 Is-a 関係とは呼べないものが多く含まれ, 乱雑なものである. 抽出した 43 のルートカテゴリと各ルートカテゴリが持つ全サブカテゴリとの関係の中で Is-a が成り立つ割合を人手により調べた結果, 約 6.5%であった. 中でも「人物」や「解剖学」などのルートカテゴリは 10,000 以上のサブカテゴリをもっており, そのほとんどが間違って記述された下位カテゴリから派生したもので占めている.

表 4 提案手法により得られた Is-a 関係の評価

ルートカテゴリ名	抽出したサブカテゴリ数	抽出したサブカテゴリの Is-a 正答率	正しい Is-a と判断されたサブカテゴリの数	再現率
有機化合物	31	1	31	1
アメリカ合衆国の州	50	1	50	1
軍人	24	1	50	0.48
銀行	3	1	35	0.085
山	84	0.95	87	0.91
アナウンサー	105	0.94	188	0.52
・・・	・・	・・	・・	・・
平均	64.60465	0.92	82.37	0.56

プロパティ定義に関しては, テンプレートで定義されていない独自のプロパティが各記事の Infobox に登場しているという結果となったが, その理由は, ユーザが Infobox を記述する際にテンプレートの項目を引用後, より具体的な項目を記述するために独自のプロパティを追加しているからであると考えられる. つまり, 編集する記事に付与するべき性質がテンプレートに定義されている性質よりも多い場合があるからである.

そこで, テンプレートに存在しないプロパティの定義域が 3.2(1)で述べた手法により得られたサブカテゴリとなり得るケースを調べた結果, いくつか当てはまる例が存在した. 例えば, テンプレート「Software」に関して, 「StepMania」という記事が持つ Infobox には, 「オンライン版」というソフトウェアのテンプレートにより定義されていないプロパティが存在する. 「StepMania」とは PC 用のオープンソース音楽ゲームであり, 属するカテゴリは「Linux 用ゲームソフト」, 「Macintosh 用ゲームソフト」, 「Windows 用ゲームソフト」と全て 3.2(1)で述べた手法により得られたサブカテゴリと一致する. したがって, 「オンライン版」の定義域として, 「ある OS 上で動作するゲーム」を持つ, というように, より具体的なプロパティ定義域抽出が可能になると考えられる.

全体的な傾向として, テンプレートで定義されていない Infobox が持つプロパティは, テンプレートが持つプロパティと同様, テンプレート名の持つ性質を正しく表している傾向が強く, ドメインに特化した性質を示す例はあまり多く見られなかった. プロパティに関するより多くの情報抽出を行うためには, 別のアプローチを考える必要もある. 例えば, テンプレートの中には「会社」と「銀行」, 「人物」と「哲学者」などテンプレート名自体に Is-a の関係が成り立っている場合も存在し, それらが持つプロパティの継承等に着目していくことも考慮していきたい.

5.4 TDO の評価実験

TDO を用いた簡単な実験と評価について述べる. 入力概念は「PHP」に関するドキュメント[*1]から Input モジュールによって得られた 55 の入力語彙を対象とする. 表 5 が TDO の出力結果である.

表 5 55 の入力語彙に対する TDO の出力結果

プロパティ	skos:related	rdfs:subClassOf	skos:prefLabel, altLabel
出力	267 facts	2 facts	14 facts
正答率	60.7 %	100 %	85.7%

表 5 が示すように, 多くの skos:related で記述される関連関係が得られた. また, skos:preflabel,altLabel で記述される同義語も高い正答率で得られた. rdfs:subClassOf で記述される Is-a 関係は, 正答率は高いものの得られた数は非常に少ない, という結果が得られた. 入力語の中に存在した Is-a 関係の数は 4 であったため, 再現率は 50%である. もし Wikipedia のカテゴリツリーを直接 Is-a 階層とみなして抽出してみると, 8 の Is-a 関係が得られるが, その正答率は 50%である. 言い換えれば, 3.2(1)で述べた Is-a 関係の抽出手法は, 正答率は高いが再現率が低い. 一方, Wikipedia カテゴリツリーを直接参照した場合, 再現率は高いが正答率は低い. したがって, より一層オントロジー構築の支援を行うためには, 我々の提案手法によって得られた Is-a 関係と Wikipedia カテゴリツリーの両者の特徴を踏まえた上で, より効果的に Is-a 関係を抽出できる手法を考える必要性がある.

6. おわりに

本研究では, Wikipedia オントロジーに基づくドメインオントロジー構築支援環境の実現と評価を行った.

Wikipedia の持つ特性に対して提案手法を適用することで Is-a 関係とプロパティ定義域を抽出し, Wikipedia オントロジーの情報量を増加させた. そして, ドメインオントロジー構築を支援するツールの開発を行い, Wikipedia オントロジーの, ドメインオントロジー構築における有用性を示した.

今後の展望としては, より多くの特定ドメインに関するドキュメントを対象にオントロジー構築を行い, Wikipedia の適用可能性を検証する. また, より洗練された Is-a 関係およびプロパティの抽出について検討していく必要がある. さらに, Wikipedia オントロジーと上位オントロジーとのマッチングを行い, Wikipedia オントロジーの質を向上させていく予定である.

また、これまで構築してきたオントロジーの成果物を、SOURCEFORGE.JP のプロジェクト内で公開している[*2].

参考文献

[手島 07]手島 拓也, 森田 武史, 和泉 憲明, 山口 高平, "日本語 Wikipedia マイニングと Folksonomy タグに基づく領域オントロジー構築支援", 第 21 回人工知能学会全国大会 1D2-5 (2007)

[桜井 08] 桜井 慎弥, 手島 拓也, 石川 雅之, 森田 武史, 和泉 憲明, 山口 高平, "汎用オントロジー構築における日本語 Wikipedia の適用可能性", 人工知能学会 第 18 回セマンティック Web とオントロジー研究会 SIG-SWO-A801-06, (2008)

[Auer 07] Sören Auer,Christian Bizer,Georgi Kobilarov,Jens Lehmann1, Richard Cyganiak,Zachary Ives: DBpedia: A Nucleus for a Web of Open Data, idelberg , ISWC/ASWC 2007, pp.722-735, LNCS 4825(2007)

[Fabian 07] Fabian M. Suchanek, Gjergji Kasneci and Gerhard Weikum "YAGO - A Large Ontology from Wikipedia and WordNet",Elsevier Journal of Web Semantics (2008)

[*1]「PHP プロ」http://www.phppro.jp/school/phpschool/vol1/

[*2]「Wikipedia Ontology プロジェクト」
http://sourceforge.jp/projects/wikipedia-ont/