2C1-NFC-2-1

日本語Wikipediaからのクラススキーマ階層の自動構築と利用

Building up Ontologies with Class Schema Hierarchy from Japanese Wikipedia and Its Use

玉川 奨[*1] 森田 武史[*2] 山口 高平[*1]
Susumu Tamagawa Takeshi Morita Takahira Yamaguchi

[*1]慶應義塾大学 Keio University
[*2]青山学院大学 Aoyama Gakuin University

Here is discussed how to build up ontologies with class schema hierarchy from Japanese Wikipedia. The ontologies includes not only class hierarchy but also class schema hierarchy that inclues classes that has properties as property domains (rdfs:domain) and property ranges (rdfs:range).

1. はじめに

大規模なオントロジーは情報検索やデータ統合において有用である．しかしながら，オントロジーの手動構築には，膨大な時間がかかり，保守や更新が困難という問題がある．そこで，近年，オントロジーの自動構築に関する研究は盛んに行われており，その情報資源として，Web 上の百科事典である Wikipedia を利用した研究は多い．Wikipedia は語彙網羅性，即時更新性に優れており，半構造情報資源であることからフリーテキストと比べてオントロジーとのギャップが小さいため，非常に優れた情報資源であるためである．しかしながら，Wikipedia を用いたオントロジー構築の多くはクラス階層構築に焦点を当てており，プロパティの定義域・値域を含めたクラススキーマ階層を構築する研究は少ない．我々はこれまでにも，日本語 Wikipedia における様々なリソース (カテゴリツリー，一覧記事，リダイレクトリンク，Infobox 等) から，概念および概念間の関係 (is-a 関係，クラスーインスタンス関係，プロパティ定義域，プロパティ値域，プロパティ関係，同義語，インスタンス間関係) を抽出し，高精度かつ大規模な汎用オントロジー (以下，日本語 Wikipedia オントロジー) を学習する手法を提案してきた [玉川 10, 玉川 11]．しかし，構築したプロパティの定義域・値域に課題があり，実用性に欠ける問題があった．そこで本稿では，プロパティ定義域・値域を洗練することで，日本語 Wikipedia からより実用的なクラススキーマ階層を備えたオントロジーの構築手法を提案し，その利用法として，近年注目されている Linked Open Data との連携を考慮した一例を示す．

2. 関連研究

DBpedia[Auer 07] は，Wikipedia の半構造情報を 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ている．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．これらは大規模なデータベースであるが，プロパティの定義域・値域については手動構築により一部定義されているのみである．

YAGO2[Johannes 10] は YAGO の知識ベースの拡張として，これまでの WordNet に Wikipedia のカテゴリを付加してオントロジーの拡張を行うだけでなく，Wikipedia と GeoNames から自空間的情報を抽出する事で，さらなるオントロジーの拡張を目指している．これら時空間的情報は wasBornOnDate や isLocatedIn といった関係を定義し，インスタンスとつないでおり，非階層関係となっている．非階層関係に着目し，時空間も含めた高度なオントロジーを構築しているが，これらの関係は手動で定義されており，プロパティの定義域や値域についても手動で定義されている．

連絡先: 玉川 奨，山口高平，慶應義塾大学理工学研究科
{s_tamagawa,yamaguti}@ae.keio.ac.jp

3. 日本語 Wikipedia オントロジーの洗練

3.1 日本語 Wikipedia オントロジー

日本語 Wikipedia オントロジーは以下の関係とタイプから構築される．本稿では定義域・値域の洗練を行うことで，クラススキーマ階層を構築し，利用法を紹介することでその有用性を示す．

1. is-a 関係 (rdfs:subClassOf)
2. クラス-インスタンス関係 (rdf:type)
3. プロパティ名とトリプル (以下のプロパティタイプを含む)
   (a) オブジェクトプロパティ (owl:ObjectProperty)
   (b) データタイププロパティ (owl:DatatypeProperty)
   (c) 対称関係プロパティ (owl:SymmetricProperty)
   (d) 推移関係プロパティ (owl:TransitiveProperty)
   (e) 関数関係プロパティ (owl:FunctionalProperty)
   (f) 逆関数関係プロパティ (owl:InverseFunctional Property)
4. プロパティ定義域 (rdfs:domain)
5. プロパティ値域 (rdfs:range)
6. プロパティ上位下位関係 (rdfs:subPropertyOf)
7. 同義語 (skos:altLabel)

3.2 クラスーインスタンス関係の洗練

日本語 Wikipedia オントロジーのクラスーインスタンス関係は一覧記事のスクレイピングにより構築している [玉川 10]．本手法によって抽出したクラス名は一覧記事名となるため，例えば，“芥川龍之介” インスタンスは “日本の小説家” クラスに属していることとなる．本手法は多くのクラスーインスタンス

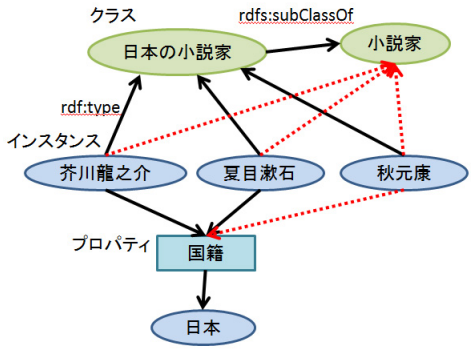

図 1: クラス—インスタンス関係の洗練の一例

関係を抽出することが可能になるが，“日本の小説家”，“アメリカの小説家” といった，クラス階層にハイブランチ構造を生じさせる問題がある．事前実験として，Wikipedia ダンプデータから抽出した 10854 の一覧記事のうち,「日本の」からはじまる記事は 624 であった．このような『国名や地域名＋格助詞「の」＋クラス名』となるクラスは多く，これらがハイブランチ構造を生む要因となっている．ハイブランチ構造によりプロパティ定義域・値域の洗練の際に，問題が生じるため，まずこの除去を行う．実際の除去の手順は次のとおりである．

1. クラス—インスタンス関係のクラス名に注目し，格助詞「の」が含まれるクラス名を抽出
2. (1) で抽出したクラスに含まれるインスタンスのうちプロパティの値が格助詞「の」の前方部となっているプロパティを抽出
3. (2) から出現頻度が少ないものを除去 (今回は 5 以下を除去した)
4. 格助詞「の」の後方部を新たなクラス—インスタンス関係として抽出
5. プロパティとプロパティの値を持たないインスタンスは抽出した関係を補完

図 1 は，本手法の一例である．“日本の小説家” クラスには “芥川龍之介”，“夏目漱石”，“秋元康” など多くのインスタンスが属している．まずクラス名の格助詞「の」に注目し，クラスに属するインスタンスのプロパティの値に “日本” が含まれるプロパティを抽出する．多くのインスタンスは “国籍” プロパティを持っており，その値は “日本” になっている．そこで，クラス名から日本を除去し，新たに “小説家” クラスのインスタンスとして定義する．さらに，これまでの日本の小説家クラスのインスタンスのうち “国籍” プロパティとその値 “日本” を持っていないインスタンス (この例では “秋元康” インスタンス) にその関係を補完する．

3.3 プロパティ定義域・値域の洗練

本手法は [森田 11] で提案した手法である．日本語 Wikipedia オントロジーの多くのプロパティ定義域はリーフとなるクラスに偏っているという問題がある．これは，プロパティ抽出をインスタンス (記事名) をベースに行っていることに起因する．インスタンスは主にリーフクラスに属するため，各記事がもつプロパティはリーフクラスに直接定義されてしまう．例えば，野球選手である “イチロー” というインスタンスは日本語 Wikipedia オントロジーにおいて “日本のプロ野球選手” というクラスに属しているため，“イチロー”(および他の日本のプロ野球選手) が持つ「国籍」や「ポジション」や「年度別打撃成績」といったプロパティは，“日本のプロ野球選手” クラスを定義域として持つ．同様に，“日本のサッカー選手” クラスのインスタンスが持つ「国籍」や「生年月日」や「ポジション」といったプロパティは “日本のサッカー選手” クラスを定義域とし，“小説家” クラスのインスタンスが持つ「国籍」「生年月日」「処女作」「受賞歴」といったプロパティは “小説家” クラスを定義域として持つ．しかし,「生年月日」や「国籍」といったプロパティは本来 “人物” クラスに定義されるべきものである．そして “人物” クラスにそれらが定義できれば，クラス階層を利用して上位クラスからプロパティ継承を用いることで，“人物” クラスの下位にあるクラスは “人物” クラスのプロパティセットを継承することができる．そこで，プロパティを持つインスタンスとクラス—インスタンス関係を用いて，各プロパティをクラスに紐付けし，親子クラス及び兄弟クラスに紐付けされたプロパティを参照する．これにより，定義域を上位クラスに統合 (リフトアップ) が可能になり，先の問題を解消する．しかしながら，本手法の問題として，is-a 階層のハイブランチ構造により，リフトアップがうまくいかないことがあった．そこで，本手法を 3.2 の手法を用いて新たに抽出した定義域・値域に適用することで，リフトアップ精度をあげるとともに，これまで行っていなかった値域にも洗練を行う．図 2 がプロパティ定義域・値域の洗練の一例である．

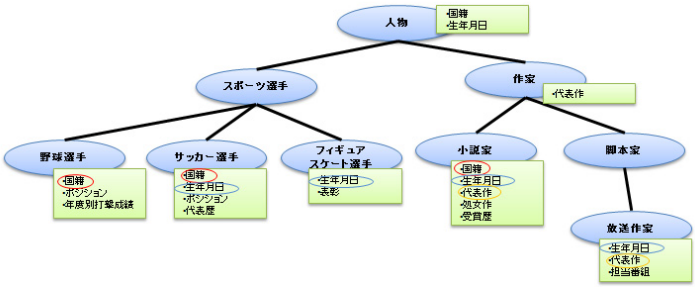

図 2: プロパティ定義域・値域の洗練の一例

4. 実験結果と考察

本実験で使用した日本語 Wikipedia オントロジーは 2012 年 1 月時点の Wikipedia ダンプデータ (jawiki-latest-pages-articles.xml)[*1]を利用し，構築したものである．データベースは MySQL，実装言語は Java 言語を用いて行った．

4.1 クラス—インスタンス関係の洗練結果と考察

Wikipedia のダンプデータから抽出した 537,810 のクラス—インスタンス関係を使用し，3.2 で提案した手法により，378 のクラスと 131,235 の関係を洗練した．表 1 に洗練したクラス名のうち関係数が多い上位 5 つのクラスを示す．最も多くインスタンスを持つクラスは “日本の漫画作品” であった．これは漫画作品のうちアニメ化されたものの多くは “放送国” プロパティとその値 “日本” をもつためである．このような国名や地名が格助詞「の」の前に来ているものは非常に多く，“日本”，“東京都”，“アメリカ合衆国” などがある．しかし，その

*1 Wikipedia ダンプデータ: http://download.wikimedia.org/jawiki/

表 1: クラスーインスタンス関係の洗練結果の一例

元のクラス名	洗練後のクラス名	関係数	属するインスタンスの一例
日本の漫画作品	漫画作品	3622	ドラゴンボール，ONE PIECE
日本の漫画家	漫画家	3592	鳥山明，手塚治虫
日本のラジオパーソナリティ	ラジオパーソナリティ	3144	山谷親平，中村鋭一
東京大学の人物	人物	2888	夏目漱石，鳩山邦夫
早稲田大学の人物	人物	2605	福原愛，江戸川乱歩

ほかにも “東京大学”，“早稲田大学” などの学校名や “平安時代”，“戦国時代” などの時代名も多い．

さらに，プロパティのトリプルとして新たに 12,051 の関係を補完した．トリプルの多くは “ビリー・ジョエル—国籍—アメリカ合衆国” や “江戸橋—都道府県—東京都” など，クラス名と同様に国名や地名が値となるものが多かった．しかし，“t.A.T.u.—ジャンル—ポピュラー音楽” や “FRONT MISSION—対応機種—プレイステーション” といったものも存在する．

しかしながら，本手法は格助詞「の」に注目しているため，それ以外のクラス名については抽出できない点や格助詞「の」を含んでいても，トリプルの値としてその前方部分が完全一致しないため取りこぼす問題などがある．例えば，“NHK のアナウンサー” クラスは格助詞「の」を持ち，“NHK のアナウンサー” クラスに属するインスタンスは “放送局” プロパティを持っているが，その値は “NHK 山口放送局” などであり，NHK と完全一致しないため，本手法では洗練できない．手法を改良し，洗練数を増やすことが今後の課題といえる．

4.2 プロパティ定義域・値域の洗練結果と考察

3.3 で提案した手法をプロパティ定義域・値域に適用した．本手法を適用することで，定義域については，“党首” プロパティの定義域が洗練前は “日本の政党”，“台湾の政党”，“宗教政党” などであったのに対し，洗練後は “政党” クラスに，“国籍” プロパティや “身長” プロパティの定義域は “人物” クラスにリフトアップしている．値域についても，定義域に比べ非常に分散しているが，“接続道路” プロパティの値域が “道路” クラスに，“付属校” プロパティの値域が “幼稚園” クラスや “小学校” クラスにリフトアップしている．しかしながら，閾値としての兄弟クラスの占める割合を変えることでリフトアップの値は大きく変わってしまう．例えば，“著作” プロパティの定義域は “小説家” クラスなどの上位クラスである “著作者” クラスが妥当であるが，兄弟クラスが定義域としてすべて含まれるものは自動構築である日本語 Wikipedia オントロジーでは少ないため，兄弟クラスのうち定義域・値域として占める割合を閾値として設定している．そのため，この値が高ければあまりリフトアップが起こらず，低ければ先の例で言うと “人物” クラスにまでリフトアップされてしまうことがある．

図 3 は兄弟クラスに占める割合を変えた際のプロパティ定義域・値域の洗練結果である．ここで兄弟クラスが占める割合を変化させると，例えば，“背番号” プロパティの定義域が “野球選手” クラスであり，“野球選手” クラスの上位クラスに “スポーツ選手”，兄弟クラスに “テニス選手” があった場合，割合が 0.5 以上であればリフトアップは行わないが，0.5 より低い場合はリフトアップが行われ，“背番号” プロパティの定義域は “スポーツ選手” となる．なお，洗練前の定義域の関係数は 67,652，値域の関係数は 54,567 であった．図を見ると，定義域の減少率が値域に比べ高いことが分かる．値域は定義域に比べ同じプロパティ名でも値の概念が広く分散していることが主な原因である．日本語 Wikipedia オントロジーでのプロパティトリプルの主語は主に記事名に対応付けされており，

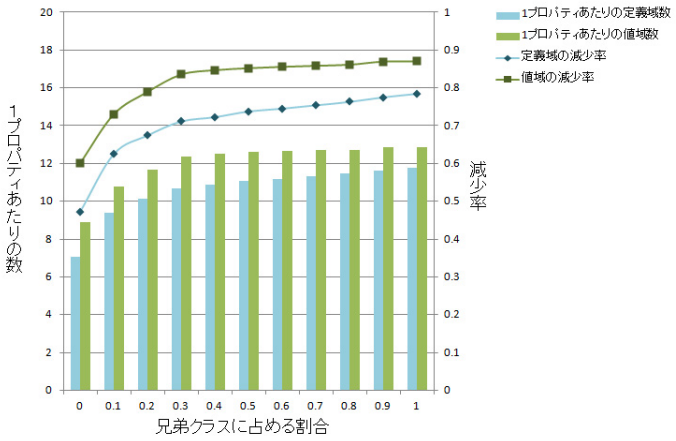

図 3: プロパティ定義域・値域の洗練結果

必ず定義域を持つのに対し，値域は記事を持たないものも多い．そのため，抽出が不十分で，クラスーインスタンス関係や is-a 関係に定義されず，概念が分散してしまっていることが考えられる．閾値を低く設定すれば定義域で 5 割程度，値域で 6 割程度，関係数を減少させる事が可能であるが，先のような問題が生じてしまう．高く設定すれば，減少率は下がってしまうが，比較的この問題は除外できる．ただし，全く無くすということはできない．例えば “著名な出身者” プロパティの値域は Wikipedia に記事がある人物はまず間違いなく著名な人物であるので，値域が “人物” クラスの下位クラス全域に分散しており，“人物” クラスにまでリフトアップしてしまう．

5. クラススキーマ階層の利用

日本語 Wikipedia オントロジーは汎用的な大規模オントロジーであり，ドメインオントロジー構築支援など，様々な利用法が考えられるが，今回は Linked Open Data の一つである XBRL Linked Open Data を利用した一例を示す．

5.1 Linked Open Data と日本語 Wikipedia オントロジー

近年セマンティック Web の研究分野では，各 Web サイトで公開されているデータベース (政府，科学，写真，音楽等) を RDF 化して連携する，Linked Open Data(LOD) が注目を集めている．残念ながら日本国内では LOD はまだ十分に普及しているとは言えないが，今後日本国内でも普及することは予想できる．各 RDF データセット間を相互にリンクするためのハブとして，英語圏では DBpedia が広く利用されているが，DBpedia は日本語固有の Wikipedia の記事には対応していないため，日本語 LOD のハブとして利用するためには十分とはいえない．日本語 Wikipedia オントロジーは，日本語固有の

表 2: プロパティと LOD・標準語彙の関連付けの一例

プロパティ名	関連先	定義域	値域
売上高	xbrlowl:NetSales	企業	リテラル
著作	dc:title,foaf:made	作家，画家	小説，漫画作品
公式ページ	foaf:homepage	人物	リテラル
緯度	geo:lat，gn:latitude	施設，企業	リテラル
所在地	gn:locatedIn	施設，企業	リテラル
駅周辺，周辺情報	gn:nearby	施設，企業	施設，企業
開発元	doap:developer	ソフトウェア	企業，人物
発売元	gr:Brand	ゲームタイトル	ゲーム会社

記事に対応しており，日本語 LOD のハブとして十分利用可能であると考える．

5.2 XBRL Linked Open Data

XBRL Linked Open Data[鈴木 11] は金融庁 EDINET の公開している XBRL を対象として，構築した会計ドメインにおける RDF モデルを用いて RDF への変換を行う事で，LOD 化している．非常に大規模な LOD であり，簡単な SPARQL クエリにより，XBRL として公開している日本国内の企業情報を取得することができる．

5.3 日本語 Wikipedia オントロジーのプロパティと LOD の関連付け

LOD はグラフ構造であり，モデルが存在しているため，日本語 Wikipedia オントロジー内の概念とモデルを直接結びつけることで，データの取得が可能である．さらに，日本語 Wikipedia オントロジーはクラススキーマ階層を持っているため，概念検索により，日本語 Wikipedia のプロパティと各 LOD のプロパティの連結が容易にできると考えられる．日本語 Wikipedia オントロジーと XBRL Linked Open Data の連結を考える際，日本語 Wikipedia オントロジーには “企業” クラスがあり，クラススキーマ階層により，“企業” クラスはどのようなプロパティがあるのかを把握できる．実際に，“企業” クラスは “売上高”，“純利益”，“従業員数”，“主要株主” など多くのプロパティを持っており，これらと XBRL Linked Open Data のモデル内での各プロパティがどういった名称であるかを把握し結びつければ，インスタンス同士を直接つなげることができ，『ゲームのハードウェアを作っている会社のここ数年の売上高の推移を示す』いったことが可能になる．実際に，日本語 Wikipedia オントロジーと LOD を用いた検索支援 TOOL WiLD(Wikipedia Linked Data Application)*2を構築し，現在公開中である．このようなプロパティの関連付けは LOD だけでなく，Dublin Core や Friend of a Friend，GeoNames といった標準語彙と関連付けすることもでき，標準語彙とプロパティを結びつけることで利便性・有用性を高める事ができるため，今後の早急な課題である．表 2 が対応付けの一例である．

6. おわりに

本稿では，日本語 Wikipedia をリソースとしてクラススキーマ階層構築手法の提案およびその評価を行った．さらに，構築したクラススキーマ階層の利用法の一例として，XBRL Linked Open Data モデルと連結し，日本語 LOD のハブとしての利用可能性を示した．クラススキーマ階層はドメインオントロジー構築支援だけでなく，LOD など既に普及しつつある技術

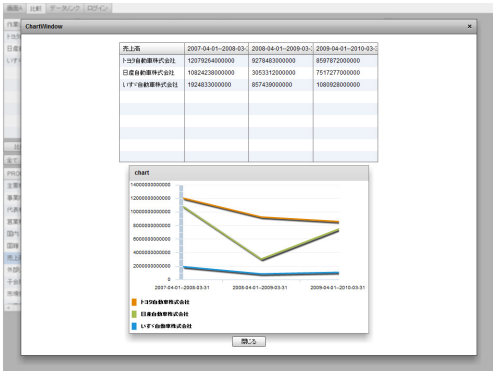

図 4: WiLD での企業情報の表示例

に利用でき，大規模なものを自動構築する事は非常に有用性が高いといえる．

今後は，日本語 Wikipedia オントロジーの規模の拡大，オントロジーの洗練について検討していく一方，実用性を考慮した提供方法を検討していく予定である．なお，日本語 Wikipedia オントロジーおよび検索システムを，SourceForge.jp*3及び日本語 Wikipedia オントロジー研究ページ*4で一部公開中であり，今後更新する予定である．

参考文献

[Auer 07] Soren Auer,Christian Bizer,Georgi Kobilarov,Jens Lehmann1,Richard Cyganiak,Zachary Ives: DBpedia: A Nucleus for a Web of Open Data ,Lecture Notes in Computer Science ,Springer Berlin / Heidelberg ,pp.722-735(2007)

[Johannes 10] Johannes Hoffart, Fabian Suchanek, Klaus Berberich, Gerhard Weikum: YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia, Research Report MPI-I-2010-5-007, Max-Planck-Institut für Informatik(2010)

[森田 11] 森田武史, 関本有佳, 玉川奨, 山口高平, ”日本語 Wikipedia からのプロパティ付きクラス階層の構築と評価”, 第 26 回セマンティックウェブとオントロジー研究会, SIG-SWO-A1103-06 (2011)

[鈴木 11] 鈴木 健太, 山口 高平, ”会計ドメインにおける RDF モデルの構築と Linked Data との連携”, 第 25 回 人工知能学会全国大会会論文集, 3E3-OS20-5(2011)

[玉川 10] 玉川 奨, 桜井 慎弥, 手島 拓也, 森田 武史, 和泉 憲明, 山口 高平, ”日本語 Wikipedia からの大規模オントロジー学習”, 人工知能学会論文誌 論文特集「2009 年度全国大会近未来チャレンジ」 Vol.25 No.5 pp.623-636 (2010)

[玉川 11] 玉川 奨, 森田 武史, 山口 高平, ”日本語 Wikipedia からプロパティを備えたオントロジーの構築”, 人工知能学会論文誌 特集論文「近未来チャレンジ」 Vol.26 No.4 pp.504-517 (2011)

*2 WiLD(Wikipedia Linked Data Application): http://wild.wikipediaontology.org/

*3 http://wikipedia-ont.sourceforge.jp/

*4 http://www.wikipediaontology.org/