1N3-OS-39b-2in1

動的インスタンスマッチング手法を用いたマッピング拡張SPARQLクエリ実行機構の拡張

A Preliminary Idea for a Mapping-enhanced SPARQL Query Execution Mechanism using Dynamic Instance Matching

足立拓也 *1　福田直樹 *2
Takuya Adachi　Naoki Fukuta

*1静岡大学情報学部情報科学科
Computer Science, Faculty of Informatics, Shizuoka University

*2静岡大学学術院情報学領域
College of Informatics, Academic Institute, Shizuoka University

Supporting heterogeneous ontologies on various LOD retrieval tasks is an important issue. There are several approaches to support the coding process of a SPARQL query for users who are unfamiliar with the used ontologies and the stored data. On the use of some ontology mapping support approaches on SPARQL-based query systems, we often assume that the users already have appropriate weighted ontology mappings for the ontologies used in the query. In this paper, we present our preliminary idea about dynamic instance matching mechanisms for mapping-enhanced SPARQL queries to widely retrieve various data from Linked Open Data. Dynamic instance mapping adaptation technique complements the used incomplete ontology mappings by dynamically detecting and adding missing mappings to include the correspondences between entities of terms in heterogeneous ontologies.

1. はじめに

Linked Open Data の検索は，Linked Open Data を読み込まれたエンドポイントに対して RDF[*1] 形式で記述されたデータを検索するクエリ言語である SPARQL[*2] に基づいてクエリを記述し発行して行う．Linked Open Data を公開しているエンドポイントの数は増加しており，Datahub[*3] に登録されているエンドポイントの情報を確認できる Web サイト [*4] では，2017 年 3 月 2 日時点で少なくとも 557 個のエンドポイントが登録されており，257 個のエンドポイントが利用可能であることが確認することができる．エンドポイントの一覧からユーザが検索したい情報が含まれているエンドポイントを推薦する研究が行われており [Adachi 16b, Hasnain 16]，このような技術を用いることによりユーザは検索したい情報が含まれているエンドポイントを調べることができる．

LOD エンドポイントで用いられるオントロジーの詳細が必ずしも十分に把握できない場合でも，重み付きオントロジーマッピング [Atencia 12] を用いた拡張 SPARQL クエリ [Fujino 14] による検索で対処できる場合がある．拡張 SPARQL クエリ実行機構では，ユーザから与えられた拡張 SPARQL クエリを標準の SPARQL クエリに変換し，オントロジーマッピングに付けられた重みを信頼度として，クエリ実行結果の順序付けおよびフィルタリングを行うことが可能である．SPARQL クエリはオントロジー中のクラスやプロパティの語彙を用いたクエリのみならず，インスタンスの語彙を用いたクエリを記述することがある．オントロジーマッピングにおいても，インスタンスマッピングを生成するための手法については，そのマッチング対象の組み合わせが膨大になることなどを考慮して特殊な手法が必要となり [Nguyen 16]，そのようにして得られたインスタンスマッピングを効果的に利用して，拡張 SPARQL クエリを実現することが，課題の 1 つとなっていた．

連絡先: 足立拓也，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，
cs13007(at)s.inf.shizuoka.ac.jp

∗1 https://www.w3.org/RDF/
∗2 http://www.w3.org/TR/rdf-sparql-query/
∗3 https://datahub.io
∗4 http://sparqles.ai.wu.ac.at

本研究では，拡張 SPARQL クエリ実行機構を動的オントロジーマッチングに対応できるように拡張した SPAIDA[足立 17] を，インスタンスマッピングを効果的に扱えるように拡張し，その拡張 SPARQL クエリを効率的に実行できるようにする．このために，クエリ記述中のインスタンスに対応するマッピングが用意したマッピングデータになかった場合でも，動的オントロジーマッチング機構 [Adachi 16a] を拡張し，インスタンスマッチングも効果的に扱えるようにするための機構の設計についても述べる．

2. 研究の背景

2.1 関係性の動的な補完機構を用いた SPARQL クエリ実行機構

我々は関係性の動的な補完機構を用いた SPARQL クエリ実行機構として SPAIDA[足立 17] の実装を進めている．SPAIDA では，動的オントロジーマッチング機構 [Adachi 16a] のような関係性の動的な補完機構を備えている．SPAIDA は，SPARQL クエリ記述を解釈し，用意されているオントロジーマッピングに対してクエリ変換に必要なマッピングを調査し，もし必要なマッピングが欠損していた場合，関係性の動的な補完機構を用いてマッピングを補い，記述されたクエリを変換し実行する．

SPARQL クエリでは，オントロジー中の語彙を用いた検索のみならず，インスタンスの語彙を用いた検索が行われる．本研究では，インスタンスマッチング [Nikolov 11] で得られる結果を用いることによって，解決の試みを行う．

2.2 インスタンスマッチング

インスタンスマッチング [Nikolov 11, Nguyen 16] は，異なる 2 つのデータセット間で同じと考えられるインスタンスを発見する技術である．Linked Data では，インスタンスマッチングは異なる 2 つの Linked Data 間で同じと考えられるインスタンスを探し出し，同一であると判断されたインスタンスのペアの間に同じであるという関係を結ぶことに利用されている．

インスタンスマッチングの難しさとして大きく 2 つの問題が挙げられる．1 つ目はインスタンスの異種性である．インスタンスを表す記述には同じ表現が使われていないことがあり，意味の曖昧さや表記のズレがある場合が

ある．例えば，DBpedia Japanese[*5] では，静岡県を表すインスタンスとして “http://ja.dbpedia.org/resource/静岡県” が用いられており，Wikidata[*6] では “https://www.wikidata.org/entity/Q131320” が用いられている．このようにマッピングを作成したいインスタンス同士だけでは，インスタンスの意味を判断することは難しく，そのインスタンスがプロパティを用いて持つ属性 (例えば，label や comment) を利用する解決手法が考えられる．

2つ目はデータの膨大さである．DBpedia Dataset 2016-04[*7] では，クラス数が 754 個に対して，インスタンス数 (Localized Instances) が 28,658,449 個存在する．このような膨大なインスタンスの中から，同一であるインスタンスのペアを探し出すためには処理時間がかかってしまうことがある．

このような問題を解決するためにインスタンスマッチングは様々な手法が提案されている [Nikolov 11, Nguyen 16]．インスタンスの異種性に対して，機械学習を用いた手法や文字列に基づく類似度計算の組み合わせなど，様々な手法で解決の試みを行われている．データの膨大さに対して，ブロッキングという手法でマッチングを行う組み合わせの削減の試みが行われている．ブロッキングはインスタンスの同一性を素早く導き出すための手法であり，すべてのインスタンスに対してマッチングを行うことを避けるために利用されている．

3. マッピング拡張 SPARQL クエリ実行機構の拡張と動的インスタンスマッチング手法の適用

本研究では，インスタンスを用いたクエリ記述をオントロジーマッピングを用いた拡張 SPARQL クエリ実行機構で実行できるようにするため，インスタンスマッチングによって得られる結果をインスタンスマッピングとして利用する．オントロジーマッピングを用いる場合と同様にインスタンスマッピングを利用するようにクエリ変換を行い，クエリを実行できるように拡張 SPARQL クエリ実行機構の拡張を行う．

また，インスタンスマッピングが用意されていなかった場合，クエリ記述中のインスタンスと検索対象の Linked Data 上のインスタンスとのマッピングを動的に補完する手法を実装する．動的オントロジーマッチング機構をインスタンスマッチング手法を適用できるように拡張する．

3.1 マッピング拡張 SPARQL クエリ実行機構の拡張

インスタンスを用いたクエリ記述をマッピング拡張 SPARQL クエリ実行機構で実行可能にするために，インスタンスマッチングの結果を用いてマッピング拡張 SPARQL クエリ実行機構上でインスタンスを用いたクエリ記述の実行方法について考える．マッピング拡張 SPARQL クエリ実行機構は Alignment Format の拡張である EDOAL[*8] で記述されたオントロジーマッピングを利用している [Fujino 14]．また，拡張 SPARQL クエリにはオントロジーマッピングに付与された信頼度を効果的に利用するための拡張構文があり，信頼度の低いマッピングのフィルタリングや信頼度の高い結果の並び替えを行うことができる．我々はインスタンスマッチングの結果をインスタンスマッピングとして EDOAL で記述し，オントロジーマッピングと同様にインスタンスマッピングを利用することで，インスタンスを用いたクエリ記述も SPAIDA 上で実行する試みを行う．

*5 http://ja.dbpedia.org
*6 https://www.wikidata.org/wiki/Wikidata:Main_Page
*7 http://wiki.dbpedia.org/dbpedia-version-2016-04
*8 http://alignapi.gforge.inria.fr/edoal.html

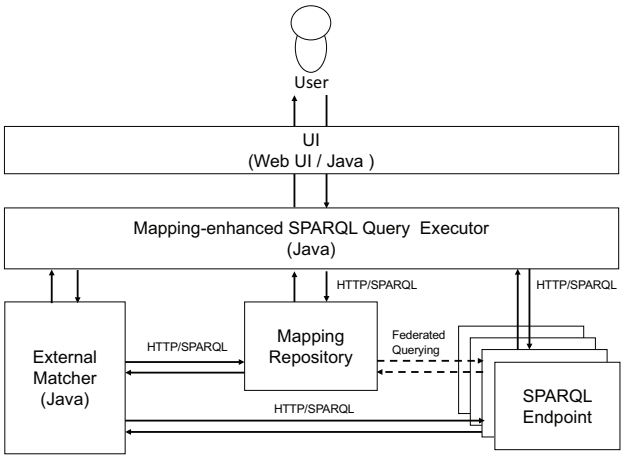

図 1: SPAIDA のアーキテクチャ

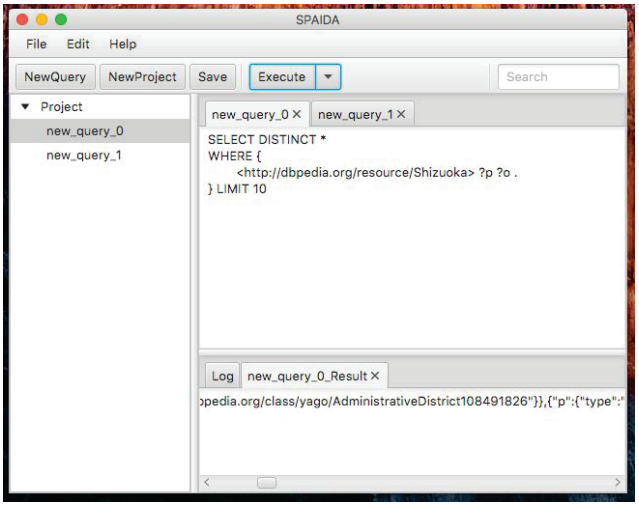

図 2: SPAIDA の表示例

SPAIDA のアーキテクチャを図 1 に示す．SPAIDA はマッピング拡張 SPARQL クエリ実行機構と動的マッチング機構を備えており，ユーザはユーザーインターフェースで SPARQL クエリや検索対象のエンドポイント，用意したマッピングなどを入力する．マッピング拡張 SPARQL クエリ実行機構では，ユーザから与えられたクエリをマッピングレポジトリにあるオントロジーマッピングやインスタンスマッピングを利用する SPARQL クエリに変換する．変換したクエリを実行した結果はユーザーインターフェースで表示する．SPAIDA は動的マッチング機構を備えており，マッピングレポジトリにクエリ記述で使用したいマッピングが欠損していた場合に動的マッチング機構を動作させることにより，マッピングの欠損を補う．

図 2 に SPAIDA の表示例を示す．プロジェクトごとに検索対象のエンドポイント等などを設定しておき，クエリを記述することで，クエリを実行することが可能となる．

3.2 動的インスタンスマッチング手法

SPAIDA では，オントロジーマッピングソフトフェアで生成されたオントロジーマッピングを用いることができ，オントロジーマッピングがなかった場合には動的にオントロジーマッ

ピングを補完することが可能な動的オントロジーマッチング機構が用意されている．動的にインスタンスマッチングを行う機構は，動的オントロジーマッチング機構を拡張し，インスタンスに対してもマッチングを行えるようにする．

本研究における動的インスタンスマッチング手法は，クエリ記述中にあるインスタンスのみに対して行い，SPAIDA 上でのクエリ実行に必要なインスタンスマッピングのみを生成することにより，必要最低限のインスタンスマッチングで実行することができる．インスタンスマッピングの出力形式は EDOAL を用いており，拡張 SPARQL クエリの特徴であるマッピングに付与された信頼度に基づく計算式に適用できるようにする．

インスタンスマッチングを行う手法は様々あり，例えば，ScLink[Nguyen 16] が挙げられる．この手法はインスタンスの膨大さから学習するデータを用意でき，ラベルのありなしを区別し，マッチングの構成を学習してからマッチングを行う．異なる 2 つのデータセットを入力することで，プロパティマッチングを行い，プロパティマッピングを生成する．プロパティマッピングを用いてブロッキングを行い，ラベル付きであるインスタンスマッピングのペア候補とラベルなしの候補の 2 つに分ける．また，プロパティマッピングを用いて類似関数を生成する．ラベル付きである候補と類似関数を用いてマッチングの構成を学習する．学習結果を用いて，ラベルなしである候補に対してマッチングするかを計算し，マッチングスコアを算出する．そのスコアを用いてフィルタリングを行い，同一であるペアを導き出す．

インスタンスマッチング手法には，マッチングを行うペア候補の探索手法やマッチングを行ったペアに付与する信頼度の計算手法がいくつか考えられ，それらの手法を組み合わせた手法も考えられる．インスタンスマッチングは様々な手法が提案されており，本機構でもそれらの手法を実装し，記述したクエリや検索対象のエンドポイントごとに手法の分析を行いたいと考える．動的マッチング機構では，そのような様々な手法の実装が行えられるように，SPARQL エンドポイントから得られる結果を用いることで，様々な手法を実装できる機構として拡張する．

例えば，インスタンスを用いた記述されたクエリは，そのインスタンスが存在する Linked Data があることを前提として記述されていると考える．動的インスタンスマッチング機構では，クエリ記述に利用した Linked Data が読み込まれているエンドポイントを入力することにより，クエリ記述中のインスタンスに関する情報を取得し，それらの情報を組み合わせたマッチング手法が考えられる．

4. 拡張した SPARQL クエリ実行機構での実行例

動的インスタンスマッチング手法の試作として，2 つの手法を実装した．1 つ目は，検索対象のエンドポイントからすべてのインスタンスを取得し，クエリ記述中のインスタンスとのマッピングを作成する手法である．1 つ目の手法はベースライン手法として実装した．2 つ目は，クエリ記述に使用したエンドポイントからクエリ記述中のインスタンスが所属しているクラスを検索し，そのクラスと検索対象の Linked Data 中のクラスとのマッピングを検索し，マッピング先のクラスに所属しているインスタンスを取得し，マッピングを作成する手法である．2 つ目の手法は用意しているオントロジーマッピングを利用し，マッチングのペア候補を削減するブロッキング手法として用いる手法である．

予備実験では，クエリ記述中のインスタンスと検索対象の Linked Data 中のインスタンスが同一の意味で扱われているペア候補に，マッピングが作成されるかの動作を確認し，用意したオントロジーマッピングがブロッキングの手法に効果的であるかを確認する．マッチングの判断手法は導入せず，すべてのペア候補に対してマッピングを作成する．実装した手法で必要となるオントロジーマッピングは，Alignment API[David 11] で WordNet を用いたマッチング手法で生成した．予備実験で用いたデータセットとして，本研究室で作成している木曽町 LOD を用いた．木曽町 LOD は 2 人の作成者が同じ文書を読み込み，独自に作成したものである．使用した LOD の詳細を表 1 に示す．

表 1: 予備実験に使用したデータセットの詳細

	R_S	R_T
Class count	190	193
Object property count	27	22
Data property count	0	17
Individual count	50	268

一例として，Listing 1 で示すクエリを実行するため，動的インスタンスマッチング機構を用いてインスタンスマッピングを作成する．Listing 1 で示すクエリは “展望環境整備” というインスタンスに関する情報を取得するクエリである．検索対象の Linked Data 中にも同様に，名前空間接頭辞が異なる記述で “展望環境整備” というインスタンスがあり，それらとのマッピングが生成されることを確認する．

```
PREFIX own: <http://www.semanticweb.org/cs13098/
    ontologies/2016/4/kiso#>
SELECT DISTINCT ?p ?o
WHERE {
 own:展望環境整備 ?p ?o .
 THRESHOLD { own:展望環境整備 > 0.5 }
 CRITERIA ?c { own:展望環境整備 * 100 }
 RANKING ?score { ?c }
} limit 10
```

Listing 1: 予備実験で使用したクエリの例

Listing 1 で示すクエリを実行した際，1 つ目の手法では，すべてのペア候補に対してマッピングを作成しているため，目的となるインスタンスマッピングが生成されることを確認した．実行時間として 1.788 秒かかった．2 つ目の手法では，オントロジーマッピングを用いたブロッキングで，目的となるインスタンスマッピングが生成されることを確認した．実行時間として 2.682 秒かかった．

正解データを作成している際に気付いた点として，必ずしもクエリ記述中のインスタンスが検索対象の Linked Data 中のインスタンスに同一のものがあるとは限らないことである．例えば，インバウンド対策として “外国語案内の推進” があるが，検索対象の Linked Data 中には同一のものはなかった．しかしながら，検索対象の Linked Data には “町営バスシステム英語表記事業” があり，該当するインスタンスは存在しないが，もしかすると類似するのではないかというインスタンスがあることを確認した．

インスタンスマッピングに付与する信頼度の計算手法を検討が必要であることがわかった．2 つ目のマッチング手法では，信頼度を用意したオントロジーマッピングに付与された重みと同じものを付与しており，インスタンスマッチングされたペアのみを確認すると信頼度が高いだろうペアが信頼度が低く与えられていることを確認した．例えば，“展望環境整備” 同士

のマッピングでは 0.413 の重みが付与されていることを確認した．

4.1 動的インスタンスマッチング手法の組み込み例

動的オントロジーマッチング機構では，Algorithm 1 で示すアルゴリズムに基づいてインスタンスマッチング手法を組み込むことができる．この手法は ScLink[Nguyen 16] を簡略化した手法であり，プロパティマッピングやブロッキングなどの処理を用意されているオントロジーマッピング，または動的オントロジーマッチング機構によって生成されたマッピングを用いて，処理時間の削減を試みる．動的インスタンスマッチング手法は，入力としてインスタンスを用いたクエリ記述 Q とクエリ記述に使用したデータセット（エンドポイント）R_S，検索対象のデータセット（エンドポイント）R_T，用意したオントロジーマッピング M_p を与え，出力としてインスタンスマッピング M_d を得る．

このような手法を組み込むため，動的マッチング機構では SPARQL クエリを利用して，クエリ記述に利用したエンドポイントや検索対象のエンドポイント，用意したマッピングデータに検索を行い，マッチングを行う候補を取得する仕組みを実装している．マッチングを行う候補をリストに格納し，そのリストに対してマッチング手法を適用することによって，動的マッピングを生成することが可能になる．

Algorithm 1 適用する動的インスタンスマッチングアルゴリズムの概要

Input: Q, R_S, R_T, M_p
Output: M_d
1: $T \leftarrow extractTerms(Q)$
2: $A \leftarrow PropertyAlignments(T, R_S, R_T, M_p)$
3: $S \leftarrow SimilarityFunctionGeneration(T, R_S, R_T, A)$
4: $C \leftarrow Blocking(T, R_S, R_T, A)$
5: $M_d \leftarrow Matching(C)$
6: return M_d

5. おわりに

本研究では，オントロジーマッピングを用いた拡張 SPARQL クエリ実行機構上でインスタンスマッピングを利用できるようにマッピング拡張 SPARQL クエリ実行機構を拡張した．また，クエリ記述中のインスタンスに対して，インスタンスマッピングが用意されていなかった場合，動的にインスタンスマッチングを行い，補助的なインスタンスマッピングを生成するための機構として，動的オントロジーマッチング機構を拡張した．マッピング拡張 SPARQL クエリ実行機構と動的マッピング拡張機構の実行例を示し，マッピング拡張 SPARQL クエリ実行機構でのインスタンスマッピングが利用できることを確認し，動的マッピング拡張機構での SPARQL クエリを用いたマッチングのペア候補を検索できる仕組みを示した．

今後の課題として，入力として与えられたクエリごとに期待されるマッピング，または期待されるマッピングを生成するマッチング手法を調査するとともに，マッピングの用意や動的マッピング機構での手法の選択や設定を効率化できる仕組みの実装が挙げられる．また，SPARQL クエリや拡張 SPARQL クエリの記述支援として，文献 [Xie 16] などの技術を用いてクエリ記述予測補完機能を検討していきたい．

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

[Adachi 16a] Adachi, T. and Fukuta, N.: Toward Better Debugging Support on Extended SPARQL queries with On-the-fly Ontology Mapping Generation, in *Proc. of The 11th International Workshop on Ontology Matching (OM2016)* (2016), (poster)

[Adachi 16b] Adachi, T., Yamada, N., and Fukuta, N.: Towards Better Query Coding Support Utilizing Ontology Mappings, in *Proc. of 1st International Workshop on Platforms and Applications for Social problem Solving and Collective Reasoning (PASSCR2016)*, pp. 96–99 (2016)

[Atencia 12] Atencia, M., Borgida, A., Euzenat, J., Ghidini, C., and Serafini, L.: A Formal Semantics for Weighted Ontology Mappings, in *Proc. of the 11th International Semantic Web Conference (ISWC2012)*, pp. 17–33 (2012)

[David 11] David, J., Euzenat, J., Scharffe, F., and Santos, dos C. T.: The Alignment API 4.0, in *Semantic Web Journal 2(1)*, pp. 3–10, IOS Press (2011)

[Fujino 14] Fujino, T. and Fukuta, N.: Utilizing Weighted Ontology Mappings on Federated SPARQL Querying, in Kim, W., Ding, Y., and Kim, H.-G. eds., *Lecture Notes in Computer Science*, Vol. 8388, pp. 331–347, Springer-Verlag (2014)

[Hasnain 16] Hasnain, A., Mehmood, Q., Zainab, e S. S., and Hogan, A.: SPORTAL: Searching for Public SPARQL Endpoints, in *Proc. of the 15th International Semantic Web Conference (Posters & Demos) (ISWC2016)* (2016)

[Nguyen 16] Nguyen, K. and Ichise, R.: ScLink: Supervised Instance Matching System for Heterogeneous Repositories, in *Journal of Intelligent Information Systems*, pp. 1–33, Springer US (2016)

[Nikolov 11] Nikolov, A., Ferrara, A., and Scharffe, F.: Data Linking for the Semantic Web, *International Journal on Semantic Web & Information Systems*, Vol. 7, No. 3, pp. 46–76 (2011)

[Xie 16] Xie, R., Liu, Z., Jia, J., Luan, H., and Sun, M.: Representation Learning of Knowledge Graphs with Entity Descriptions, in *Proc. of the 30th AAAI Conference on Artificial Intelligence(AAAI2016)*, pp. 2659–2665 (2016)

[足立 17] 足立 拓也, 福田 直樹 : SPAIDA: 関係性の動的な補完機構を用いた SPARQL クエリ実行機構の提案, 第 41 回セマンティックウェブとオントロジー研究会 (SIG-SWO-041) (2017)