SIG-SWO-045-03

Wikipedia記事からの中間RDFグラフとDBpediaトリプルの抽出

Extracting RDF Graphs and DBpedia Triples from Wikipedia Articles

末木 顕人[1*] 兼岩 憲[1]
Kent Sueki[1] Ken Kaneiwa[1]

[1] 電気通信大学大学院 情報理工学研究科 情報・ネットワーク工学専攻
[1] Department of Computer and Network Engineering, Graduate School of Informatics and Engineering, The University of Electro-Communications

Abstract: RDF トリプルの自動抽出タスクには，これまで主に Wikipedia の Infobox のような半構造化データが利用されてきた．一方で，近年ではテキストのような非構造化データからの抽出が注目されている．本研究では，Wikipedia 記事本文のコンテンツを構造化データとして活用するため，自然言語文の係り受け関係と述語項構造に基づく中間 RDF グラフ及びその生成手法を提案する．中間 RDF グラフは他のデータセットとリンクすれば意味的な連携が可能で，記事本文中の内容に基づいた検索や情報抽出に応用できる．その応用例として，DBpedia Japanese と連携し中間 RDF グラフ上の構造を DBpedia 上のプロパティに対応付け，Wikipedia 記事本文から DBpedia の RDF トリプルを抽出する．

1 はじめに

セマンティック Web において Linked Open Data (LOD) は Web 上で計算機によって解読可能なデータを公開・共有するプロジェクトである．そうした機械可読なデータを記述する枠組みに RDF (Resource Description Framework) があり，リソース（情報の単位）間の関係を，主語，述語，目的語の三つ組（RDF トリプル）で表現する．

LOD プロジェクトの 1 データセットとして DBpedia がある [1]．DBpedia では Wikipedia 記事における Infobox やカテゴリ構造などの構造化・半構造化データから，半自動的に RDF データが生成されている．また，特に日本語版 Wikipedia を対象とした取組みには，DBpedia Japanese プロジェクト [2] や，玉川ら [3] の日本語 Wikipedia オントロジーがある．しかし，これらの取組みでは記事本文などの自然言語文からデータ抽出されず，獲得できる有用な知識が残されている．

これに対して，自然言語文から RDF を生成する取組みが研究されている．Exner ら [4] は英語版 Wikipedia 記事の自然言語文から述語項構造を抽出し，Named Entity Linking (NEL) によって固有表現である項を DBpedia 上のリソースと対応付ける．さらに，ブートストラップ法によって述語に対応する DBpedia プロパティを獲得して，RDF トリプルへ変換する．TEXT2LOD[5] は，Fillmore の格文法に基いて，自然言語文中における各文節の格（Subject, Action, Object, Time, Location）をルール及び Conditional Random Fields (CRF) によって決定し，RDF 形式の語間関係を抽出する．

しかし，現状では構造化データとしての日本語文の活用や，大規模 LOD の名前空間に対応した RDF データの (半) 自動生成が十分に研究されていない．加えて，Exner らの手法では述語項構造からそのまま RDF トリプル化するため，体言による連体修飾を利用した「リュック・ベッソンはレオンの監督である.」のような表現から，RDF データを獲得できない．

本研究では．日本語版 Wikipedia 記事の本文から係り受け関係と述語項構造に基づく中間 RDF グラフを生成する方法を提案する．その中間 RDF グラフから，語及び語間関係をそれぞれ DBpedia データ中のリソース及びプロパティと対応付け，DBpedia の RDF トリプルを自動生成する．提案手法では，日本語文の述語項構造だけでなく係り受け関係を利用し，名詞間の関係を含む語間関係を抽出する．

*連絡先：電気通信大学大学院 情報理工学研究科
情報・ネットワーク工学専攻
東京都調布市調布ケ丘 1-5-1
E-mail: sueki@sw.cei.uec.ac.jp

2 準備

2.1 RDF グラフ

まず，RDF グラフ及び RDF トリプルについて定義する [6]．RDF は，URI 参照の集合 U，空ノードの集合 B，及びリテラルの集合 L から構成される．主語 $s \in U \cup B$，述語 $p \in U$，目的語 $o \in U \cup B \cup L$ に対して，RDF トリプルは以下の 3 つ組で表される．

$$(s, p, o)$$

また，主語や目的語をリソース，述語をプロパティ，目的語をプロパティ値と呼ぶ．このとき，RDF トリプルの有限集合 $G \subseteq (U \cup B) \times U \times (U \cup B \cup L)$ を RDF グラフと呼ぶ．RDF グラフは主語を始点，目的語を終点，述語を有向辺とした有向グラフであり，リソース間の関係性を記述できる．RDF グラフ内の述語 p をもつすべての主語と目的語の組集合を $G(p) = \{(s, o) | (s, p, o) \in G\}$ とする．

2.2 述語項構造と語間関係

本研究では，RDF トリプルを抽出するために，日本語の文章における係り受け関係及び述語項構造を利用する．文書 $S = (st_1, st_2, \ldots, st_N)$，文 $st_i = (\phi_1^i, \phi_2^i, \ldots, \phi_n^i)$，文節 ϕ_j^i $(i = 1, 2, \ldots, N)$ $(j = 1, 2, \ldots, n)$ に対して，係り受け関係を次のように定義する．

定義 1 (係り受け関係) 文 st において文節 $\phi, \psi \in st$ 間に係り受け関係があり，ϕ が表層格 c を伴って ψ に係っているとき，$\phi \prec_c \psi$ と書く．

自然言語文における述語には動詞や形容詞，名詞 (但し，「だ」，「である」，「です」を後ろにもつ) があり，項には動作や状態の主語や，目的語，場所，時間を表す名詞などがある．述語項構造を次のように定義する．

定義 2 (述語項構造) 文書内の文 $st_1, st_2 \in S$ において述語の文節 $\phi \in st_1$ と ϕ の項の文節 $\psi \in st_2$ とが述語項構造を作り，ψ が表層格 c をもつとき，$PAS_{\phi,c}(\psi)$ と書く．

また，係り受け関係及び述語項構造を利用して，文節間の関係構造を示す語間関係を定義する．

定義 3 (語間関係) 文書 S に含まれる任意の述語の文節 ϕ 及び項の文節 ϕ_1, ϕ_2 に対して，$\phi_1 \prec_c \phi$ または $PAS_{\phi,c}(\phi_1)$，かつ $\phi_2 \prec_{c'} \phi$ または $PAS_{\phi,c'}(\phi_2)$ であるとき，ϕ_1, ϕ_2 は語間関係 $A^{\phi}_{c,c'}(\phi_1, \phi_2)$ をもつという．

このとき，$A^{\phi}_{c,c'}$ を語間関係 $A^{\phi}_{c,c'}(\phi_1, \phi_2)$ の関係ラベルと呼ぶ．関係ラベル空間 $\mathcal{A}$ と文字列空間 Σ^* に対して，語間関係を文字列に変換する全単射写像をラベル関数 ℓ $(\ell : \mathcal{A} \to \Sigma^*)$ とする．

3 中間 RDF グラフ

本章では，日本語版 Wikipedia 記事本文から表層格に基づいた中間 RDF グラフ (構造化データ) を生成する．中間 RDF グラフは主に文中に現れる文節 ϕ の実体に相当するリソース（語リソース $r(\phi)$ と呼ぶ）と，それらを関係付けるプロパティ（表層格）によって各文節・語の格構造を表現する．語リソースは「文書名-文番号-文節番号-文節の表層」という形式で URI を決定し，表 1 の 4 つのプロパティ語彙で関係を記述する．また，文書自体のリソースは「文書名」に基づいて URI を決定する．

生成元文章と中間 RDF グラフ及び DBpedia との関係の例を図 1 に示す．例えば，以下は st_2 が文，$\phi_1^2, \phi_2^2, \phi_3^2$

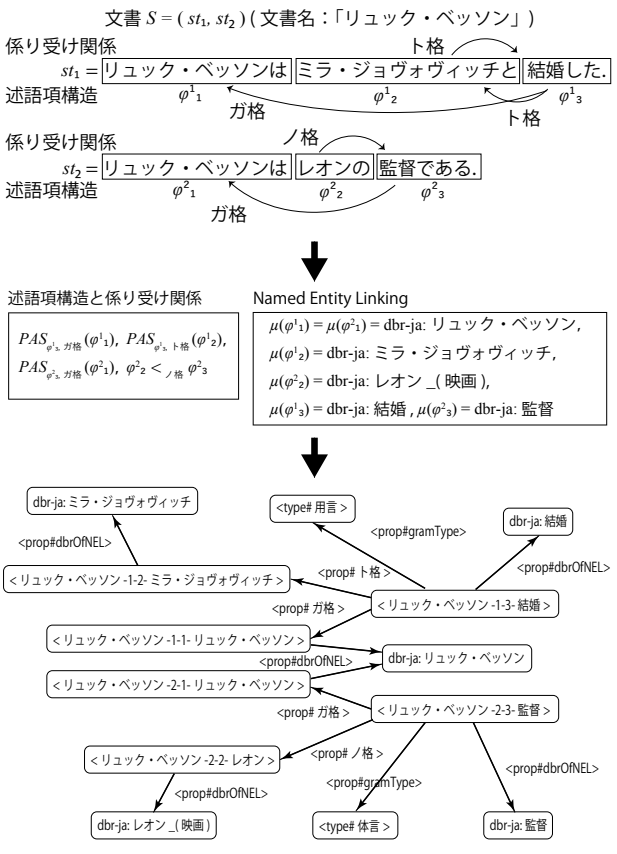

図 1: 中間 RDF グラフの生成例

がそれぞれ文節で，特に ϕ_3^2 は述語の文節，ϕ_1^2 は ϕ_3^2 の項の文節である．

st_2 ＝ リュック・ベッソンはレオンの監督である．
ϕ_1^2 ＝ リュック・ベッソンは

表 1: 中間 RDF グラフのプロパティ語彙

プロパティ	プロパティの用途
prop#{ 表層格 }	文節 ϕ，ψ が，ψ が持つ任意の表層格 c に対して $PAS_{\phi,c}(\psi)$ もしくは $\psi \prec_c \phi$ であるとき，語リソース $r(\phi)$，$r(\psi)$ に対してトリプル $(r(\phi), <\text{prop\#c}>, r(\psi))$ を作る．
prop#contentOf	語リソースをドメインとして，目的語の文書に語リソースが所属していることを示す．
prop#gramType	「体言」か「用言」それぞれ $<$ *type*#体言 $>$，$<$ *type*#用言 $>$ として持つ．
prop#dbrOfNEL	語リソース $r(\phi)$ を主語として，NEL 処理によって文節 ϕ に対応する URI $\mu(\phi)$ が獲得できた場合には $\mu(\phi)$ を目的語としたトリプル，得られなかった場合には空ノード _: *surface*(ϕ) を目的語としてトリプルを作る．

$$\begin{aligned}\phi_2^2 &= \text{レオンの}\\ \phi_3^2 &= \text{監督である}\end{aligned}$$

これにより，係り受け関係 $\phi_2^2 \prec_{ノ格} \phi_3^2$ と述語項構造 $PAS_{\phi_3^2,\,ガ格}(\phi_1^2)$ を獲得し，次のトリプルを生成する．

$$(r(\phi_3^2), <\text{prop\#ガ格}>, r(\phi_1^2))$$
$$(r(\phi_3^2), <\text{prop\#ノ格}>, r(\phi_2^2))$$

また，NEL 処理によって文節と DBpedia Japanese 名前空間上の URI

$$\begin{aligned}\mu(\phi_1^2) &= \text{dbr-ja: リュック・ベッソン}\\ \mu(\phi_2^2) &= \text{dbr-ja: レオン\_(映画)}\\ \mu(\phi_3^2) &= \text{dbr-ja: 監督}\end{aligned}$$

を獲得し，各 $i \in \{1,2,3\}$ に対して以下のトリプルを生成する．

$$(r(\phi_i^2), <\text{prop\#dbrOfNEL}>, \mu(\phi_i^2))$$

ただし，述語の文節 ϕ_3^2 に対して URI $\mu(\phi_3^2)$ が得られない場合は，空ノードを用いて以下のトリプルを生成する．このとき，*surface*(ϕ_3^2) は文節の表層である．

$$(r(\phi_3^2), <\text{prop\#dbrOfNEL}>, \text{\_: } surface(\phi_3^2))$$
$$(\text{\_: } surface(\phi_3^2), \text{rdfs: label}, surface(\phi_3^2))$$

こうして生成された中間 RDF グラフに対して SPARQL クエリを発行し，文書中のコンテンツを意味的に検索できる．また，プロパティ `prop#dbrOfNEL` によって DBpedia（やその他の LOD）中のリソースと語とを対応付ければ，図 2 のように，既存の知識ベースを中間 RDF グラフにより補完できる．

4 DBpedia RDF トリプルの抽出

本章では，中間 RDF グラフを活用して日本語版 Wikipedia 記事から DBpedia RDF トリプルを抽出する．まず，中間 RDF グラフで表現された語間関係をトリプル形式 (候補トリプル) に変換する．NEL 処理によって，文節 ϕ_1，ϕ_2 に対応する名前空間 NS 上のリソース URI $\mu(\phi_1)$，$\mu(\phi_2)$ が得られる．このとき，語間関係 $A^{\phi}_{c,c'}(\phi_1,\phi_2)$ は以下の候補トリプルに変換される．

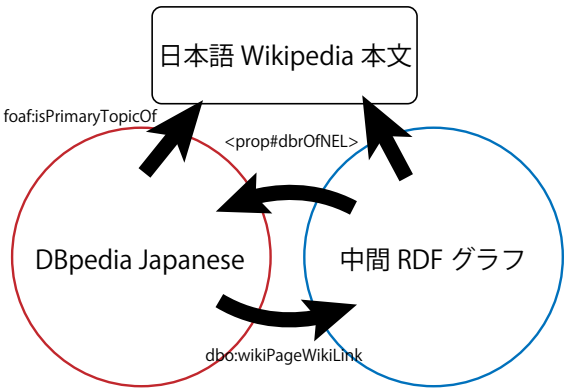

図 2: 中間 RDF グラフと DBpedia との連携

$$(\mu(\phi_1), \ell(A^{\phi}_{c,c'}), \mu(\phi_2))$$

続いて，語間関係から特に PAS 関係及び役割関係を定義し，SPARQL クエリを使ってその 2 つの関係の候補トリプルを抽出する．さらに，抽出された候補トリプルから DBpedia 名前空間へのオントロジーマッピング及び評価実験について述べる．

4.1 PAS 関係の抽出

PAS 関係は 2 組の述語項構造から構成される，「A が B を V する」を意味する語間関係である．

定義 4 (PAS 関係) 語間関係 $A^{\phi}_{c,c'}(\phi_1,\phi_2)$ をもつ文節 ϕ_1，ϕ_2 において，特に $PAS_{\phi,\,ガ格}(\phi_1)$ かつ $PAS_{\phi,c}(\phi_2)$ $(c \neq ガ格)$ であるとき，ϕ_1，ϕ_2 は *PAS* 関係をもつという．

例えば，図 1 の文 st_1 から以下の PAS 関係及び対応する候補トリプルを獲得できる．

$$\begin{aligned}st_1 &= \text{リュック・ベッソンはミラ・ジョヴォヴィッチと結婚した．}\\ &\rightarrow A^{結婚}_{ガ格,\ ト格}(\text{リュック・ベッソン, ミラ・ジョヴォヴィッチ})\\ &\rightarrow (\text{dbr-ja : リュック・ベッソン}, \ell(A^{結婚}_{ガ格,\ ト格}),\\ &\qquad \text{dbr-ja : ミラ・ジョヴォヴィッチ})\end{aligned}$$

中間 RDF グラフから PAS 関係の候補トリプルを抽出するためにソースコード 1 の SPARQL クエリを実行する．クエリ変数 ?*verb*，?*subject*，

ソースコード 1: PAS 関係の候補トリプルを抽出する SPARQL クエリ

```
1  CONSTRUCT { ?dbr_subject CONCAT(?case1, '+',
       ?label_verb, '+', ?case2) ?dbr_object }
2  WHERE {
3    ?verb ?case1 ?subject .
4    ?verb ?case2 ?object .
5    ?verb <prop#gramType> <type#用言> .
6    ?subject <prop#dbrOfNEL> ?dbr_subject .
7    ?object <prop#dbrOfNEL> ?dbr_object .
8    ?verb <prop#dbrOfNEL> ?dbr_verb .
9    ?dbr_verb rdfs:label ?label_verb .
10   FILTER (?case1 = <prop#ガ２格
         > || ?case1 = <prop#ガ格>) .
11   FILTER (?case2 != <prop#ガ２格
         > && ?case2 != <prop#ガ格
         > && ?case2 != <prop#未格>) .
12   FILTER (?dbr_subject != ?dbr_object) .
13 }
```

?*object*，?*case*1，?*case*2，?*label_verb*，?*dbr_subject*，?*dbo_object*，?*dbo_verb* は，それぞれ述語を含む用言の実体，その主語の実体，その目的語の実体，用言と主語とを結ぶ格，用言と目的語とを結ぶ格，用言の表層，主語に相当する DBpedia リソース，目的語に相当する DBpedia リソース，用言に相当する DBpedia リソースである．9 行目は主語と用言とを結ぶ格をガ格またはガ２格に制限し，10 行目は目的語と用言とを結ぶ格をガ格，ガ２格，未格以外に制限し，11 行目は主語と目的語の DBpedia リソースが異なることを指定する．このクエリにより，文 st_1 から以下の候補トリプルを抽出する．

(dbr-ja : リュック・ベッソン, ガ格+結婚+ト格,
dbr-ja : ミラ・ジョヴォヴィッチ)

4.2 役割関係の抽出

役割関係は「A は B の V である」を意味し，それぞれ 1 組の述語項構造と係り受け関係とから，特殊な語間関係により構成される．

定義 5 (役割関係) 語間関係 $A^{\phi}_{c,c'}(\phi_1, \phi_2)$ をもつ文節 ϕ_1，ϕ_2，及び体言の文節 ϕ において，特に $PAS_{\phi,\ \text{ガ格}}(\phi_1)$ かつ $\phi_2 \prec_{\text{ノ格}} \phi$ であるとき，ϕ_1，ϕ_2 は役割関係をもつという．また，ϕ を役割と呼ぶ．

例えば，図 1 の文 st_2 から以下の役割関係及び対応する候補トリプルを獲得できる．

$$
\begin{aligned}
st_2 \quad &= \quad \text{リュック・ベッソンはレオンの監督である．} \\
&\rightarrow \quad A^{\text{監督}}_{\text{ガ格},\ \text{ノ格}}(\text{リュック・ベッソン}, \text{レオン}) \\
&\rightarrow \quad (\text{dbr-ja} : \text{リュック・ベッソン}, \ell(A^{\text{監督}}_{\text{ガ格},\ \text{ノ格}}), \\
&\qquad \text{dbr-ja} : \text{レオン（映画）})
\end{aligned}
$$

PAS 関係と同様に，ソースコード 2 の SPARQL クエリにより役割関係の候補トリプルを抽出する．ク

ソースコード 2: 役割関係の候補トリプルを抽出する SPARQL クエリ

```
1  CONSTRUCT { ?dbr_subject CONCAT(?case1, '+',
       ?label_role, '+', ?case2) ?dbr_object }
2  WHERE {
3    ?role ?case1 ?subject.
4    ?role ?case2 ?object.
5    ?subject <prop#dbrOfNEL> ?dbr_subject.
6    ?object <prop#dbrOfNEL> ?dbr_object.
7    ?role <prop#gramType> <type#体言>.
8    ?role <prop#dbrOfNEL> ?dbr_role .
9    ?dbr_role rdfs:label ?label_role .
10   FILTER (?case1 = <prop#ガ２格
         > || ?case1 = <prop#ガ格>).
11   FILTER (?case2 = <prop#ノ格>).
12   FILTER (?dbr_subject != ?dbr_object).
13 }
```

エリ変数 ?*role*，?*subject*，?*object*，?*case*1，?*case*2，?*role_verb*，?*dbr_subject*，?*dbo_object*，?*dbo_role* は，それぞれ，役割を示す名詞の実体，その主語の実体，その目的語の実体，役割と主語とを結ぶ格，役割と目的語とを結ぶ格，役割の表層，主語に相当する DBpedia リソース，目的語に相当する DBpedia リソース，役割に相当する DBpedia リソースである．9 行目は主語と用言とを結ぶ格をガ格またはガ２格に制限し，10 行目は目的語と用言とを結ぶ格をノ格に制限し，11 行目は主語と目的語の DBpedia リソースが異なることを指定する．これにより，文 st_2 から以下の候補トリプルを抽出する．

(dbr-ja : リュック・ベッソン, ガ格+監督+ノ格,
dbr-ja : レオン（映画）)

4.3 オントロジーマッピング

日本語版 Wikipedia 記事本文から得られた候補トリプルにオントロジーマッピングを適用して，DBpedia の RDF トリプルを生成する．まず，文書 S から生成されたすべての候補トリプルの集合を T とする．このとき，$T(\ell(A^{\phi}_{c,c'}))$ は，述語 $\ell(A^{\phi}_{c,c'})$ をもつすべての候補トリプルの主語と目的語の組集合である．

次に述語 $\ell(A^{p}_{c,c'})$ を名前空間 NS 上のプロパティにマッピングする．NS 上の RDF グラフ G が与えられた

とき，G 内のトリプルから，述語 $\ell(A^{\phi}_{c,c'})$ と同じ主語と目的語の組を最も多くもつ NS 上のプロパティ p_{mapped} を得る．

$$p_{mapped} = \arg\max_p(|T(\ell(A^{\phi}_{c,c'})) \cap G(p)|)$$

さらに，このマッピングにクラス制約を導入する．まず，リソース r のオントロジークラス $C(r)$，マッピングした述語 p_{mapped} の主語と目的語の組集合 $G(p_{mapped})$ を得る．ただし，r のクラスが得られないとき，$C(r) = *$ として任意のクラスを許容する．これにより，主語と目的語のクラスが最頻出する組合せが以下で求まる．

$$(C_s, C_o) = \arg\max_{(C_1,C_2)}(|\{(s,o) \in T(\ell(A^{\phi}_{c,c'})) \cap G(p_{mapped}) | C(s) = C_1, C(o) = C_2\}|)$$

ここで，オントロジーマッピング辞書 D を次のように定義する．

$$D(\ell(A^{\phi}_{c,c'}), C_s, C_o) = p_{mapped}$$

よって，語間関係 $A^{\phi}_{c,c'}(\phi_1, \phi_2)$ の候補トリプルから NS 上の RDF トリプル

$$(\mu(\phi_1), D(\ell(A^{\phi}_{c,c'}), C(\mu(\phi_1)), C(\mu(\phi_2))), \mu(\phi_2))$$

へマッピングできる．

例として，以下のような PAS 関係の候補トリプルの述語 $\ell(A^{\text{結婚}}_{\text{ガ格, ト格}})$ = ガ格+結婚+ト格 の DBpedia Japanese 名前空間へのオントロジーマッピングを考える．

$$\begin{aligned}&(\text{dbr-ja}:\text{リュック・ベッソン},\ \ell(A^{\text{結婚}}_{\text{ガ格, ト格}}),\\ &\ \text{dbr-ja}:\text{ミラ・ジョヴォヴィッチ})\\ =\ &(\text{dbr-ja}:\text{リュック・ベッソン},\\ &\ \text{ガ格+結婚+ト格},\\ &\ \text{dbr-ja}:\text{ミラ・ジョヴォヴィッチ})\end{aligned}$$

この述語をもつすべての主語と目的語の組では，`dbo: spouse` (配偶者の関係を示す DBpedia オントロジープロパティ) が最も高頻度で出現する．この場合，「ガ格+結婚+ト格」を述語にもつ候補トリプルの主語と目的語の組集合 $T(\ell(A^{\text{結婚}}_{\text{ガ格, ト格}}))$，DBpedia Japanese の RDF グラフ G より，

$$\begin{aligned}p_{mapped} &= \arg\max_p(|T(\ell(A^{\text{結婚}}_{\text{ガ格, ト格}})) \cap G(p)|)\\ &= \text{dbo: spouse}\end{aligned}$$

となる．さらに，主語と目的語のクラスの中で最頻出の組が以下で求まる．

$$\begin{aligned}(C_s, C_o) &= \arg\max_{(C_1,C_2)}(|\{(s,o) \in T(\ell(A^{\text{結婚}}_{\text{ガ格, ト格}}))\\ &\quad \cap G(p_{mapped}) | C(s) = C_1, C(o) = C_2\}|)\\ &= (\text{dbo: Person}, \text{dbo: Person})\end{aligned}$$

表 2: クラス制約と抽出精度・抽出量との関係

主語と目的語のクラス制約条件	抽出量 (個)	精度 (%)
両方 $*$ を許容	3,716	34.0
どちらか一方の $*$ を許容	3,142	61.0
両方 $*$ を非許容	1,214	72.0

これにより，オントロジーマッピング辞書 D を以下のように得る．

$$\begin{aligned}&D(\ell(A^{\text{結婚}}_{\text{ガ格, ト格}}), \text{dbo: Person}, \text{dbo: Person})\\ &\qquad = \text{dbo: spouse}\end{aligned}$$

辞書 D を全ての候補トリプルに順次適用して，最終的に，以下の DBpedia Japanese 名前空間の RDF トリプルを得る．

$$\begin{aligned}&(\text{dbr-ja}:\text{リュック・ベッソン},\ \text{dbo: spouse},\\ &\ \text{dbr-ja}:\text{ミラ・ジョヴォヴィッチ})\end{aligned}$$

5 実験

5.1 実験方法

日本語版 Wikipedia から抽出した中間 RDF グラフ及び DBpedia Japanese のデータを用いて，Wikipedia 記事本文を構造化した DBpedia RDF トリプルを抽出する．候補トリプルからのオントロジーマッピングには，DBpedia Japanese[1] 中のオントロジープロパティを利用したトリプル，及びオントロジークラスの記述に相当するトリプルを採用した．この RDF データより 97,479 個のリソースを無作為に選択し，当該リソースに対応する Wikipedia 記事 97,479 件（計 1,542,462 文）が含む文章をダンプデータ[2] から抽出し，中間 RDF グラフを生成した．この際，日本語構文解析器として KNP[3] を，NEL ツールとして jawikify[4][7] を利用した．実験で生成した RDF トリプルから無作為に 100 トリプルを選択し，抽出元文の文意に合っているか否かを人手で判定して精度評価を行った．

5.2 RDF トリプルの抽出

オントロジーマッピングにおけるクラス制約の厳密さを変えて，RDF トリプルの抽出精度及び抽出量を計測した結果を表 2 に示す．最も抽出量が多いのはクラス制約が主語と目的語ともに $*$ (任意のクラス) を許容する場合で，最も精度が高いのはともに $*$ を非許容な場合であった．これは，クラス制約が厳しいほど精度

[1] `http://ja.dbpedia.org/dumps/20160407/`
[2] `https://dumps.wikimedia.org/jawiki/20180101/`
[3] `http://nlp.ist.i.kyoto-u.ac.jp/?KNP`
[4] `https://github.com/conditional/jawikify`

表 3: 各語間関係の抽出した RDF トリプルの精度とエラー率

	精度 (%)	エラー率 (%) 主語	述語	目的語
PAS 関係	54.0	19.6	91.3	10.9
役割関係	89.0	45.5	81.8	27.3

表 4: 各語間関係の RDF トリプルの抽出量

	抽出量	新規獲得トリプル数
PAS 関係	2,881	1,853
役割関係	241	64
合計	3,122	1,917

が高くなり，抽出量が減少することを示す．以降の実験では抽出の精度と量のバランスの良い，主語と目的語どちらかの $*$ を許容する条件を採用した．

次に，PAS 関係と役割関係とに対応する RDF トリプルの精度及び生成量を比較する．表 3 の精度は下式で算出し，エラー率は不正解トリプル全体のうちの要素の割合である．

$$\text{精度} \quad = \quad \frac{\text{正解トリプル数}}{\text{正解トリプル数} + \text{不正解トリプル数}}$$

加えて，表 4 に各語間関係におけるトリプルの抽出量と，DBpedia にない新規に獲得したトリプル数の比較を示す．PAS 関係と比較して役割関係の精度が非常に大きい．一方で役割関係のトリプル抽出量は少なく，新規に獲得したトリプル数の割合も少ない．

5.3 オントロジーマッピング辞書の生成

表 5，表 6 に PAS 関係と役割関係のオントロジーマッピングで得られた高頻度の結果をそれぞれ示す．表左は，候補トリプルの生成時に主語及び目的語に課すクラス制約である． この結果より，語間関係ラベルから DBpedia のプロパティへ適切にマッピングできていることがわかる．例えば

ガ格+移籍+ヘ格

は，ある人物がスポーツチームへ移籍することを記した文でよく出現する語間関係の関係ラベルで，`dbo:team` へマッピングしている．また，PAS 関係のマッピングでは目的語クラス制約の多くが任意のクラスになっている．

5.4 考察

実験結果によると，主語と目的語のクラス制約がともに $*$ を非許容とすると RDF トリプルの抽出量が $1,214$ と大幅に少ない．この原因は，DBpedia Japanese のリソースに関するクラスの記述不足が考えられる．実際，表 5 及び 6 の結果は任意のクラスを多く含み，クラスの記述不足を示唆する．例えば，語間関係ラベル

ガ格+受賞+ヲ格

では，dbo: award の値域として任意のクラスよりも dbo: Award が期待される．

また，表 3 によると，抽出したトリプルの述語におけるエラー率が非常に高く，主な原因がオントロジーマッピング辞書の誤りだった．例えば，クラス制約に $*$ を含む次のようなオントロジーマッピング辞書

$$D(\ell(A^{\text{務める}}_{\text{ガ格, デ格}}), \text{dbo: Person}, *) = \text{dbo: team}$$

は，次の候補トリプルにも適用される．

$$(\text{dbr-ja : ヒラリー・クリントン}, \ell(A^{\text{務める}}_{\text{ガ格, デ格}}), \text{dbr-ja : アメリカ合衆国})$$

しかし，アメリカ合衆国はスポーツチームではないため不適当である．よって，トリプルの抽出量を維持しつつクラスの記述を補ってマッピングの精度を高くする課題が残る．

加えて，語間関係の違いも精度や抽出量に大きく影響していた．PAS 関係と比較して役割関係の RDF トリプルの精度は高いが抽出量は少なかった．この要因は，語間関係それぞれの情報量の違いが挙げられる．例えば，役割関係 $A^{\phi}_{c,c'}(\phi_1, \phi_2)$ の関係ラベルにおける文節 ϕ は体言であるから，それ単体で意味を成しやすい．一方，PAS 関係 $A^{\phi}_{c,c'}(\phi_1, \phi_2)$ では述語の文節 ϕ として「なる」や「ある」など，述語と表層格だけでは意味を特定しづらい．故に，PAS 関係が前述のオントロジーマッピングで多くの失敗を引き起こす．この対策としては，語間関係の定義を拡張して構成要素を増やす，もしくは PAS 関係を細分化して情報量を増加させる方法が考えられる．

6 関連研究

本章では，自然言語文からの RDF トリプル抽出に関する関連研究を示す．本研究と類似のアプローチに Exner ら [4] の手法がある．Exner らは英語版 Wikipedia において DBpedia の解析対象になっていなかった記事本文から，語間関係として述語項構造を抽出し，その関係を DBpedia 上のリソース及びプロパティと対応付けて RDF トリプルを生成する手法を提案した．

しかし，当手法は PropositionBank[8] で定義された述語項構造のアノテーションに依存しており，日本語文への適用は困難である．また，抽出対象の関係を述

表 5: PAS 関係における高頻度のマッピング

主語クラス制約	語間関係ラベル	目的語クラス制約	マッピング p_{mapped}
dbo:Person	ガ格+移籍+ヘ格	*	dbo:team
dbo:Person	ガ格+率いる+ヲ格	dbo:Organisation	dbo:party
dbo:Person	ガ格+移籍+ニ格	*	dbo:team
dbo:Person	ガ格+受賞+ヲ格	*	dbo:award
dbo:Person	ガ格+在籍+ニ格	dbo:Organisation	dbo:team

表 6: 役割関係における高頻度のマッピング

主語クラス制約	語間関係ラベル	目的語クラス制約	マッピング p_{mapped}
dbo:Place	ガ格+支流+ノ格	dbo:Place	dbo:riverMouth
dbo:Organisation	ガ格+子会社+ノ格	dbo:Organisation	dbo:owner
dbo:Person	ガ格+子+ノ格	dbo:Person	dbo:parent
dbo:Organisation	ガ格+政党+ノ格	*	dbo:country
dbo:Person	ガ格+アナウンサー+ノ格	dbo:Organisation	dbo:affiliation

語項構造（本研究の PAS 関係に該当）に制限しており，体言による連体修飾の関係などには対応できない．当手法に対して，本研究は日本語テキストからの抽出を可能とし，係り受け関係と述語項構造を統合した中間 RDF グラフにより，多様な関係性の獲得をしやすくする．

7 おわりに

本研究では，日本語版 Wikipedia 記事本文のコンテンツを構造化データとして活用するために，中間 RDF グラフの生成方法を提案した．中間 RDF グラフは日本語文中の係り受け関係及び述語項構造を統合した構造をもち，他の RDF データセットと連携して，日本語文からの意味的検索に役立つ．更に，この中間 RDF グラフを応用して，Wikipedia の内容に基づく DBpedia Japanese の RDF トリプルを自動抽出し，評価実験を行った．

今後の課題としては，`owl:sameAs` プロパティを利用して DBpedia Japanese のオントロジークラスを補強する拡張がある．また，Wikipedia 記事以外の日本語文を対象としたり，Wikidata[5] などの他のデータセットに対するオントロジーマッピングの適用，役割関係以外の語間関係を中間 RDF グラフから抽出することを検討している．

参考文献

[1] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, and C. Bizer. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. In the Journal Semantic Web, Vol. 6, No. 2, pp. 167–195, 2015.

[2] 加藤文彦. DBpedia の現在:リンクトデータ・プロジェクト. 情報管理, Vol. 60, No. 5, pp. 307–315, 2017.

[3] 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, Vol. 26, No. 4, pp. 504–517, 2011.

[4] P. Exner and P. Nugues. Entity Extraction: From Unstructured Text to DBpedia RDF Triples. In Proceedings of the Web of Linked Entities Workshop (WoLE 2012), pp. 58–69, 2012.

[5] 川村隆浩, 大須賀昭彦. TEXT2LOD ～テキスト情報の LOD 化に向けた Web API の開発～. 人工知能学会論文誌, Vol. 31, No. 1, pp. 1–8, 2015.

[6] 兼岩憲. RDF と RDF スキーマの推論. 人工知能学会論文誌, Vol. 26, No. 5, pp. 473–481, 2011.

[7] S. Zhou, K. Matsuda, R. Tian, N. Okazaki, and K. Inui. A Pipeline Japanese Entity Linking System with Embedding Features. In Proceedings of 30th Pacific Asia Conference on Language, Information and Computation (PACLIC 30), pp. 267–276, 2016.

[8] M. Palmer, D. Gildea, and P. Kingsbury. The proposition bank: An annotated corpus of semantic roles. Computational linguistics, Vol. 31, No. 1, pp. 71–106, 2005.

[5] `https://www.wikidata.org/`