Vol.2011-IFAT-101 No.4
Vol.2011-NL-200 No.4
2011/1/28

Linked Dataから得られるリソース間関係に着目した情報拡張手法の提案

大西　可奈子[†1]　　小　林　一　郎[†1]

本研究では，Linked Data を用いてユーザの興味に関連する情報を提供する，新しい情報拡張手法を提案する．具体的には，ユーザが興味を持つ文章の内容を最もよく表わしている語と，それに強い関連を持つ語について，Linked Data を用いて二語間の関係を説明する情報をユーザに提示する．さらに，取得した情報が持つ関係と同じ関係を持つ情報を Linked Data を用いて取得し，それを拡張情報としてユーザに提示する．これにより，ユーザの興味に基づいて情報を拡張し，提供することを可能にする．

An Information Enhancement Technique Focusing on the Relationship between Resources obtained from Linked Data

Kanako Onishi[†1] and Ichiro Kobayashi[†1]

In this paper, we propose an information enhancement technique. The technique provides a user with the information that she or he interests. Concretely, the technique provides a user with the information about the relationship between two words through Linked Data, one of those is the word which expresses the content of the sentences the user interests and the other word has a strong relation with the former word. Furthermore, it obtains the information which has the same relation to the obtained information through Linked Data, and then the obtained information is provided to a user as a piece of enhanced information. Through this process, based on a user's interest, we can enhance information and provides it to the user.

†1 お茶の水女子大学大学院　人間文化創成科学研究科　理学専攻
Ochanomizu University, Graduate School of Humanities and Sciences Reorganization Proposal, Advanced Sciences

1. はじめに

近年，大容量かつ多様化する Web ドキュメントをどのようにして有効に扱うかが大きな課題となってきている．そこで，この問題の有効的な解決方法に成り得ると考えられるメタデータやセマンティック・ウェブの技術が，現在改めて注目されている．セマンティック・ウェブは 1998 年ごろに Tim Berners-Lee 氏によって提唱された技術 1) であり，従来の HTML では伝えきれなかった，語彙の意味なども記述できる XML、XML Schema、RDF、RDF Schema、OWL などから成る．セマンティック・ウェブが注目を浴びる中，セマンティック・ウェブ技術のひとつとして Tim Berners-Lee 氏が新たに提唱したのが Linked Data2)3) である．これまでのセマンティック・ウェブが，クラスすなわちオントロジーに焦点を当てられていたのとは対照的に，Linked Data ではインスタンスすなわちリソースに焦点が当てられている．Linked Data は，外部から参照可能な RDF データで，個々の RDF ファイルは唯一つの URI を持ち，その RDF ファイル内にはひとつのインスタンス（事物）に関する記述がされている．それらのインスタンスが属するクラスは，オントロジーによって定義される．従来のドキュメントは読み物であり，クリックすることにより別のページに飛びリンクをたどることができるが，ハイパーリンクによって結ばれていない情報を得ることはできない．そのようなデータ間を関係に基づくリンクで結ぶことにより，異なる領域にわたった質問を可能にするなど，個別のデータがより有用なものとなった．

主要な Linked Data のいくつかとして，すべての国の地理情報，および 800 万の地名を Linked Data で記述した Geonames4)，音楽のメタデータデータベースである MusicBrainz5)，概念辞書である WordNet6)，Dbpedia7) などがある．DBpedia は，Wikipedia から構造化された情報を抽出し，その情報を Web で利用可能な RDF の形にして提供しているものである．抽出した語彙には、それぞれ URI が与えられており，その URI に語彙の概念や，固有名詞が持つ情報などが記述されている．例えば，Wikipedia の Tokyo に対応するデータは次の URI，“http://dbpedia.org/resource/Tokyo”で表現され，そこには”Tokyo”の街としての概念だけでなく，同時に国土や人口などの数値データも記述されている．Wikipedia の膨大な語彙から作成されるため，その数は英語版でおよそ 150 万と膨大である．本研究では，このようにして日々作られている膨大な Linked Data の有効な活用法として，ユーザが興味を持った文章を対象に Linked Data を用いてユーザに提供する情報を拡張する手法を提案する．なお，本研究では幅広い分野の文章を対象としているため，知識獲得には語彙の豊富な DBpedia を利用する．また，SPARQL クエリによる知識

Vol.2011-IFAT-101 No.4
Vol.2011-NL-200 No.4
2011/1/28

獲得の実行速度の問題から現在は DBpedia のみを利用しているが，将来的に実行速度の問題などが解消されれば，すべての Linked Data を対象にしたいと考えている．

2. 関 連 研 究

検索エンジンの開発において Linked Data を利用した多くの研究がなされている．そのようなものに，Swoogle8)，Watson9)，SWME10)，Sindice11) などが挙げられる．検索エンジン以外では，コンテンツを Linked Data と結び付けるアノテーション技術により，検索精度を従来よりも高める研究も数多く報告されている．例えば BBC は，BBC のコンテンツを Linked Data で記述し，DBpedia や MusicBrainz とリンクさせるシステムを開発している 12)．対象コンテンツをビデオコンテンツに特化したものとして，彼らはビデオデータのための意味検索を容易にするための手法を提案した 13)．ここで Waitelonis らは，アカデミックビデオサーチ Yovisto14) のサーチインデックスを適切な DBpedia リソースにマッピングする方法を，八つのステップに纏めた．また Bernhard らは，ユーザが簡易メモを入力すると，システムが潜在的に関連があると思われる DBpedia リソースをランク付けして提示するビデオアノテーション手法を提案した 15)．さらに，セマンティックデータのマッシュアップアプリケーションとして Linked Data を使う事例も報告されている．例えば，Aastrand らは，意味的な背景知識を DBpedia を通して利用することにより，コンテンツをより容易に分類する手法 16) を提案した．また，DBpedia Mobile17) は，GPS 情報を用いて携帯にユーザの位置情報に加えて，その位置情報に関連する情報を DBpedia から取得しラベルやアイコンで表示する．地理情報と Linked Data を用いた研究には他にも，沿岸エリアと失業率などの統計変数との間にある関係を分析した研究 18) 等がある．これらの研究はいずれも実際に記述されている物・事（リソース）に対して，Linked Data を用いて，そのリソースに関する追加情報を取得することにより，検索を容易にしたり情報拡張を行うものである．本研究では，あるリソースに対する追加情報を取得するだけでなく，ある文章に記述されている二つのリソース間にある「関係」を取得し，ユーザに提示する．加えて，それらと同じ関係を持つ別のリソースを取得し，その情報もユーザに提示することにより，ユーザの興味に沿った情報拡張を行うことを目的とする．

3. 情報拡張手法

3.1 システム概観

本研究で提案する情報拡張手法を実現するシステムの概観を**図 1** に示す．

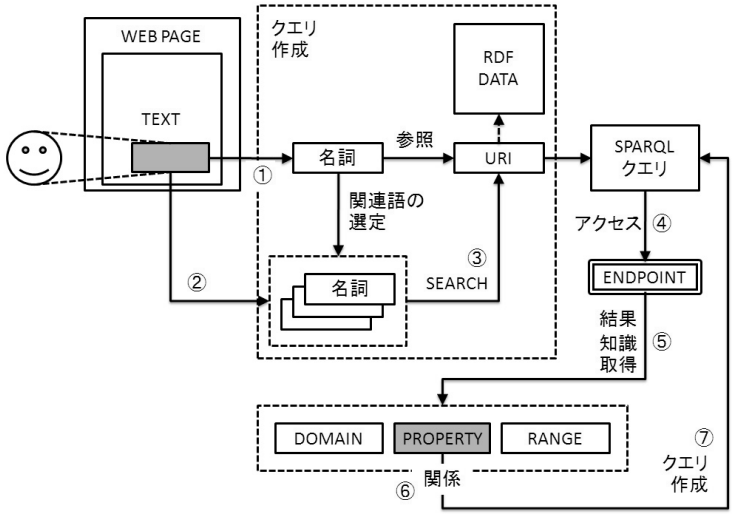

図 1 システム概観
Fig. 1 System overview

システムの処理の流れを以下に説明する．初めにユーザが Web ページに記述されたある文章の中のある一部分に興味を持ったと仮定する．この時システムはまず，その文章の選択された範囲の内容を最もよく表わしていると考えられる名詞をひとつ抽出する（①）．このような名詞を「内容語」と呼ぶことにする．なお，この名詞の抽出方法は次節にて詳細を述べる．この名詞は一つ目のリソースとなり，今後そのリソースに対する URI が参照される．例えば，その名詞が「Tokyo」だった場合，Tokyo に与えられている URI が指す RDF 内に記述されているデータを解析し，ユーザが興味をもった一部分より得られる知識の抽出を行う．

しかし，RDF は必ず Domain, Property, Range の三つ組で記述されているため，リソースが一つ決まっただけでは特定の知識を抽出することはできない．そこで，その名詞に対してのその他の名詞の関連の強さ，および重要度を求める（②）．重要度の算出方法は 3.3 節にて詳細を述べる．この時，重要度を数値化したものが最大となる語を内容語の「関連語」と呼ぶ．ここで，内容語と関連語の関係を見つけ

Vol.2011-IFAT-101 No.4
Vol.2011-NL-200 No.4
2011/1/28

るために，内容語 URI が指す RDF データの中に関連語を含む知識を探す（③）．

その知識にしたがって，システムは RDF クエリ言語 SPARQL19) クエリを自動作成し，エンドポイントを通して Linked Data にアクセスし（④），土台語と関連語に関係する知識の抽出を行う（⑤）．ここで抽出される知識は RDF 言語で記述されているため，必ず「関係」を持つ（⑥）．本研究では，その「関係」を持つもので，かつ内容語を含まないものを拡張情報とする．ここでもシステムは，SPARQL クエリを自動作成し（⑦），再度エンドポイントを通して Linked Data にアクセス，必要な知識の抽出を行う．

なお，本研究では対象文書に含まれる名詞のみを利用した．これは，本研究では最終的に RDF ファイルを扱うため，RDF 記述と親和性の低い形容詞などは利用せず，リソースやプロパティ名となりやすい名詞のみを利用することが，本目的を達成する上で単純かつ効果的であると考えたためである．

3.2 内容語抽出

本研究では,「何度も繰り返し言及される単語は重要な単語である」という仮説を立て，ユーザが興味を持った文章を最もよく表わしている単語は，文章中において何度も繰り返し言及されている単語と考える．また,「重要な単語は一箇所に偏らず文章全体に現れる」という仮説を立て，文章中に頻出し，かつ文章全体に万遍なく出現している名詞が，ユーザが興味を持った文章 D を最もよく表わしている名詞であると考える．

そこで，文章 D に含まれる名詞集合を $N = \{n_1, n_2, ..., n_i\}$ とする．N は語の重複を許さない名詞の集合とする．また，ある名詞 の文章 D における出現頻度を $f_D(n)$，ある名詞 の文章 D における出現位置を $pos(n) = \{p_1, p_2, ..., p_{f_D(i)}\}$ と表わす時，名詞 n のバラつきの程度 $W(n)$ を不偏分散を用いて以下の式で求める．

$$W(n) = y = \frac{1}{(f_D(n)-1)^{\alpha}} \sum_{i=1}^{n} (|p_{i+1} - p_i| - \bar{p})^2$$

これは,「単語 n が出現する間隔が，文章 D を単語 n の出現回数で割ったもの（単語間の距離の平均）に近ければ近いほど，ある単語 n が文章 D 上で最も均等に散らばっている」と考えられるためである．ここで，$\frac{1}{f_D(n)-1}$ を α 乗しているのは，本研究の趣旨から出現回数を強く考慮するためであり，α は経験的に 3 とする．また，$\bar{p}$ は以下の式で表わされる．

$$\bar{p} = \frac{X}{f_D(n)}$$

ここで，X は文章 D に含まれる全単語数を表わす．すなわち，$\bar{p}$ は n が最も均等に分散した場合の単語間距離を表わしている．$W(n)$ は n が広範囲かつ均等に出現している時，最小となる．したがって，$W(n)$ を最小とする n が文章 D を最もよく表わしている名詞だと言える．本研究では，そのような名詞 n が文章 D の「内容語」となる．ここで，名詞集合 n を $W(n)$ に基づいて昇順に並べ替えたものを新たに $N = \{n_1, n_2, ..., n_i\}$ とする．

3.3 関連語抽出

次に，3.2 節で抽出した内容語に対して関連語を見つける手法を示す．関連語とは，文章 D 中に出現し，内容語と強い関連を持ち，かつ重要だと思われる名詞のことである．本研究では，内容語との関連の度合いは相互情報量を用いて表し，重要度は単語の出現回数の平方根を用いて表す．したがって，内容語に対しての関連語らしさ $K(n_k, n_l)$ は以下のように表せる．

$$K(n_k, n_l) = I(n_k, n_l) \times \sqrt{f_D(n_l)}$$

ここで，$I(n_k, n_l)$ は内容語 n_k と関連語候補 n_l の相互情報量であり，次の式で求められる．

$$I(n_k, n_l) = \log \frac{p(n_k, n_l)}{p(n_k)p(n_l)} \quad (n_k \neq n_l)$$

ここで，

$$p(n_k, n_l) = \frac{f_D(n_k, n_l)}{X}, \quad p(n_k) = \frac{f_D(n_k)}{X}, \quad p(n_l) = \frac{f_D(n_l)}{X}$$

また，X は文章 D の名詞総数を表し，$f_D(n_k, n_l)$ は n_k と n_l が同時に一文に出現する頻度を表す．これは,「ある二つの単語が一文に同時に現れた時，その二単語は強い関係によって結ばれている可能性がある」と考えられるためである．したがって，$I(n_k, n_l)$ が大きいほど n_k と n_l は強い関係で結びついているとみなせる．

ここで，ある内容語 n_k について，その他のすべての名詞の $K(n_k, n_l)$ を求め，$K(n_k, n_l)$ について降順に並び変えた集合を N' とし，$K(n_k, n_l)$ を最大にする名詞 n_l を関連語と呼ぶ．

3.4 関 係 抽 出

これら二つの名詞を基に，関係の抽出を行う．まず，行列の要素 $T_{ij} (i = 0, 1, 2; 0 \leq j \leq m-1)$ が $\{0, 1\}$ をとる以下の行列を定義する．

Vol.2011-IFAT-101 No.4
Vol.2011-NL-200 No.4
2011/1/28

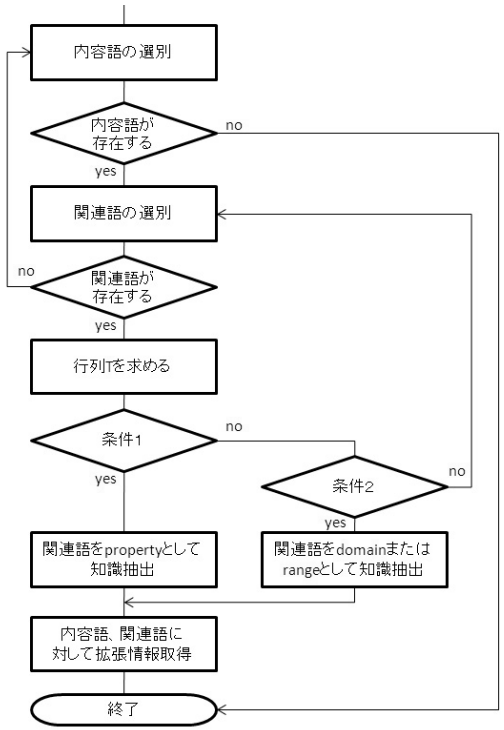

図 2 知識抽出アルゴリズム
Fig. 2 Knowledge extraction algorithm

$$T_{ij} = \begin{bmatrix} T_{00} & T_{01} & \cdots & T_{0m-1} \\ T_{10} & T_{12} & \cdots & T_{1m-1} \\ T_{20} & T_{23} & \cdots & T_{2m-1} \end{bmatrix}$$

この行列は，内容語 n の RDF データに記述されている関連語を含むすべての三つ組に対して，domain に関連語を含む場合は $T_{0j} = 1$，property に関連語を含む場合は $T_{1j} = 1$，range に関連語を含む場合は $T_{2j} = 1$ とする．

関係の抽出は行列 T を使って**図 2** に示す流れで行われる．

この時，行列 T が「$(\sum T_{1j} \geq \sum T_{0j}) \wedge (\sum T_{1j} \geq \sum T_{2j})$（図 2 における**条件 1**）」を満たす時，すなわち抽出した関連語が property として多く記述されている場合，内容語をリソース，関連語を含むプロパティを関係として，知識の抽出を行う．

または，行列 T が，「$(\sum T_{0j} \geq \sum T_{1j}) \wedge (\sum T_{0j} \geq \sum T_{2j}) \wedge (\sum T_{0j} < \alpha)$ または，$(\sum T_{2j} \geq \sum T_{0j}) \wedge (\sum T_{2j} \geq \sum T_{1j}) \wedge (\sum T_{2j} < \alpha)$（図 2 における**条件 2**）」を満たす時，すなわち抽出した関連語がリソースとして多く記述されている場合，内容語をリソース，関連語をもう一つのリソースとし，その二つのリソースの間にある関係を抽出する．ここで，関連語が文章の特徴を表わさない単語だった場合，どの文章にも頻出する単語と考えられるため $\sum T$ は非常に大きな数値になると想定され，これは期待される単語の抽出がなされない．したがって，予備実験の結果を踏まえて，現在のところ α を経験的に 40 としたが，α は対象領域に依存して決められる．

$\sum T = 0$ となった場合は，3.2 節で求めた $K(n_k, n_l)$ が n_l の次に大きい名詞について行列 T を求め直す．すべての n について $\sum T = 0$ となった場合は，3.1 節で求めた $W(n)$ が，n の次に小さい名詞を新たに文章 D の内容語とし，N' を求め直す．

3.5 クエリの作成

関係知識を Linked Data を用いて抽出するための QUERY は自動で作成される．なお本研究では DBpedia へは，RDF クエリ言語 SPARQL を使い，エンドポイント `http://dbpedia.org/sparql` からアクセスする．

前節において関連語を含む property が解析対象であると判断された場合，ある文章から抽出される知識は「**リソース R（内容語）に対して，関連語を含む関係（property）P をもつリソース**」になる．したがって，文章 D から抽出できる知識は，SPARQL のコマンドで表現すると，

```
SELECT ?hasValue WHERE { ⟨R⟩ ⟨P⟩ ?hasValue }
```

または，

```
SELECT ?isValueOf WHERE { ?isValueOf ⟨P⟩ ⟨R⟩ }
```

で求められる．

前節において関連語を含む domain または range が解析対象であると判断された場合，文章 D から抽出される知識は「**リソース R（内容語）が、その他の関連語を含むリソース R' との間に持つ関係（property）**」である．したがって，文章 D から抽出できる知識は，

Vol.2011-IFAT-101 No.4
Vol.2011-NL-200 No.4
2011/1/28

```
SELECT ?property WHERE { ⟨R⟩ ?property ⟨R'⟩ }
```

または，

```
SELECT ?property WHERE { ⟨R'⟩ ?property ⟨R⟩ }
```

で求められる．

また，上記二つのクエリから抽出された知識が持つ関係を P とする時，新たに抽出される情報は「**関係 P を持つ domain と range の組**」である．したがって，文章 D に対して拡張できる情報は，

```
SELECT ?isValue ?hasValue WHERE { ?isValue ⟨P⟩ ?hasValue }
```

で求められる．これにより，選択文章から抽出した知識がもつ関係を保存した，新たな知識を抽出することができる．

3.6 絞り込み

3.5 節において結果が一定以上の場合は，リソースのカテゴリによって絞り込みを行う．例えば，**図 3** において，ある内容語を Resource1 とする時，関連語より関係 P が取得できるとする．この時，拡張できる情報は前節で述べたように，「関係 P を持つ domain と range の組み」である．SPARQL クエリによって Linked Data から得られた結果リソースを，図 3 で示す R 群（グループ α とグループ β の和）とする．この R 群が多量であった場合，Resource1 のクラスを取得．図 3 においては Resource1 はクラス B に所属するため，結果リソースの内クラス B に所属するグループ α のみが抽出される．

3.7 表　　示

抽出された知識を表現する際，表示には対象プロパティP のラベルを用いる．ラベルの取得には以下のクエリを用いた．

```
SELECT ?hasValue WHERE { ⟨P⟩ rdfs:label ?hasValue }
```

ラベルの定義がなされていないプロパティについては，プロパティの名前をそのまま利用した．このラベル L を用いて，抽出した知識以下の形式で表示する．

```
L of isValue(domain) is hasValue(range).
```

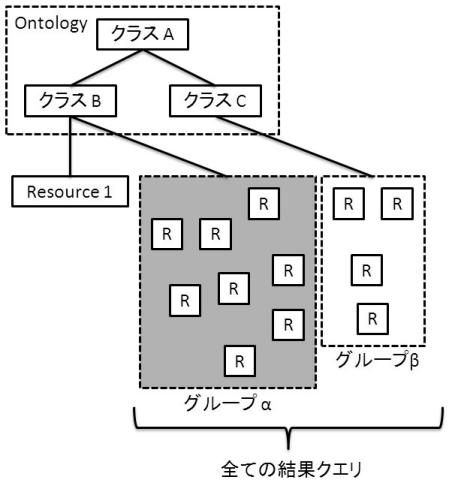

図 3　カテゴリによる絞り込み
Fig. 3　Narrow down based on category

4. ケーススタディ

Wikipedia の Supermarine_Spitfire の項目[⋆1] において，**図 4** に示す反転部分がユーザの興味を持った箇所とする．この時，内容語を抽出するために $W(n)$ を求めと，$W(n)$ 値の上位 10 項目は**表 1** のようになる．

したがって，分散値 $W(n)$ を最小とする「Spitfire」が選択部分の内容語となる．なお，DBpedia の Spitfire の項目[⋆2] には Spitfire についての具体的な記述はなく，代わりに以下の下線部に示すように，Supermarine_Spitfire を参照するようにとの記述がなされている．

```
⟨rdf:Description rdf:about="http://dbpedia.org/resource/Spitfire"⟩
⟨dbpprop:redirect
rdf:resource="http://dbpedia.org/resource/Supermarine_Spitfire"/⟩
⟨/rdf:Description⟩
```

⋆1 http://en.wikipedia.org/wiki/Supermarine_Spitfire
⋆2 http://dbpedia.org/data/Spitfire.rdf

Vol.2011-IFAT-101 No.4
Vol.2011-NL-200 No.4
2011/1/28

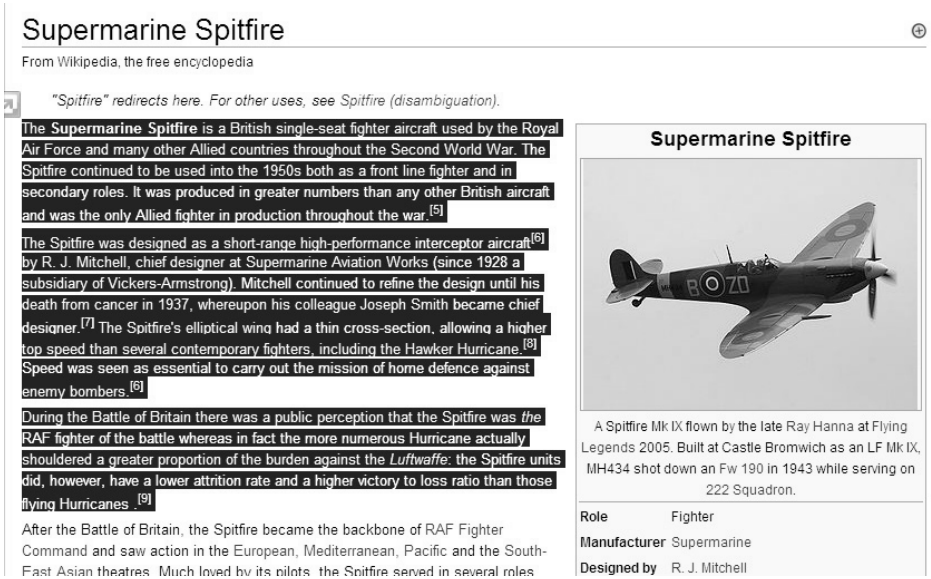

図 4 Wikipedia の Supermarine_Spitfire のページ
Fig. 4 Supermarine_Spitfire's page of Wikipedia

よって，内容語が「Spitfire」の時，参照 RDF ファイルは Supermarine_Spitfire.rdf[⋆3] となる．この時，関連の強さおよび重要度を表わす $K(n_k, n_l)$ 値の上位 10 項目は**表 2** のようになる．

したがって，内容語「Spitfire」に対する関連語は，$K(n_k, n_l)$ を最大とする「fighter」となる．以下に，Supermarine_Spitfire.rdf 内の三つ組に，関連語「fighter」を含むもの一部を掲載する．

⋆3 http://dbpedia.org/data/Supermarine_Spitfire.rdf

表 1 選択部の $W(n)$ 値の上位 10 項目
Table 1 $W(n)$ top 10

名詞	出現回数	$W(n)$ 値
spitfire	6	4.305
fighter	5	20.18
hurricane	3	266.24
aircraft	3	366.06
supermarine	2	506.25
british	2	3192.25
war	2	4032.25
designer	2	5402.25
chief	2	5402.25
battle	2	7656.25

表 2 選択部の $K(n_k, n_l)$ 値の上位 10 項目
Table 2 $K(n_k, n_l)$ top 10

内容語	名詞	$K(n_k, n_l)$ 値
spitfire	fighter	7.440362423916648
spitfire	hurricane	5.447489751833141
spitfire	aircraft	5.447489751833141
spitfire	r	5.160012695170857
spitfire	raf	4.649187071404865
spitfire	air	4.243721963296702
spitfire	battle	4.041012868497678
spitfire	speed	4.041012868497678
spitfire	supermarine	4.041012868497678
spitfire	british	4.041012868497678

```
⟨rdf:Description rdf:about="http://dbpedia.org/resource/No._79_Squadron_RAAF"⟩
⟨dbpprop:aircraftFighter xmlns:dbpprop="http://dbpedia.org/property/"
rdf:resource="http://dbpedia.org/resource/Supermarine_Spitfire"/⟩⟨/rdf:Description⟩

⟨rdf:Description rdf:about="http://dbpedia.org/resource/RAF_Fighter_Command"⟩
⟨dbpprop:aircraftFighter xmlns:dbpprop="http://dbpedia.org/property/"
rdf:resource="http://dbpedia.org/resource/Supermarine_Spitfire"/⟩⟨/rdf:Description⟩

⟨rdf:Description
```

Vol.2011-IFAT-101 No.4
Vol.2011-NL-200 No.4
2011/1/28

rdf:about="http://dbpedia.org/resource/No._302_Polish_Fighter_Squadron"⟩
⟨dbpprop:equipment xmlns:dbpprop="http://dbpedia.org/property/"
rdf:resource="http://dbpedia.org/resource/Supermarine_Spitfire"/⟩
⟨/rdf:Description⟩

⟨rdf:Description rdf:about="http://dbpedia.org/resource/No._457_Squadron_RAAF"⟩
⟨dbpprop:aircraftFighter xmlns:dbpprop="http://dbpedia.org/property/"
rdf:resource="http://dbpedia.org/resource/Supermarine_Spitfire"/⟩
⟨/rdf:Description⟩

なお，該当 RDF が関連語「fighter」を含まない場合は，図 2 のアルゴリズムに従って $K(n_k, n_l)$ 値が次に高い「aircraft」を新たに関連語とする．内容語リソース RDF がすべての関連語を一度も含まない場合，$W(n)$ 値が「Spitfire」の次に高い fighter が内容語となり，fighter に対しての $K(n_k, n_l)$ 値を計算し直し，以下同じ手順が続く．

この例では，内容語リソースの RDF データが関連語を含んでいたため，行列 T は以下のように表わせる．

$$T = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

この行列は，3.4 節で述べた条件 $(\sum T_{1j} \geq \sum T_{0j}) \wedge (\sum T_{1j} \geq \sum T_{2j})$ を満たす．したがって，内容語をリソース，関連語を含むプロパティ
http://dbpedia.org/ontology/aircraftFighter
を関係として，知識の抽出を行う．

すなわち，この文章から抽出される知識は「**Supermarine_Spitfire に対して関係** aircraftFighter **をもつ domain または range**」である．この時 SPARQL クエリは，

```
SELECT ?isValue
WHERE { ?isValue ⟨http://dbpedia.org/ontology/aircraftFighter⟩
⟨http://dbpedia.org/resource/Supermarine_Spitfire⟩ }
```

または，

```
SELECT ?hasValue
WHERE { ⟨http://dbpedia.org/resource/Supermarine_Spitfire⟩
⟨http://dbpedia.org/ontology/aircraftFighter⟩ ?hasValue }
```

となる．

これらのクエリはエンドポイントを通り，いくつかの結果を得る．ここで，表示のためにプロパティのラベルを求める．SPARQL クエリは以下の通りである．

```
SELECT ?hasValue
WHERE { ⟨http://dbpedia.org/ontology/aircraftFighter⟩
rdfs:label ?hasValue }
```

これにより，プロパティ aircraftFighter のラベル「aircraft fighter」を取得する．結果，以下の知識（抽出されたものの一部を掲載）がユーザに提示される．

aircraft fighter of No._303_Polish_Fighter_Squadron is Supermarine_Spitfire.
aircraft fighter of No._453_Squadron_RAAF is Supermarine_Spitfire.
aircraft fighter of No._457_Squadron_RAAF is Supermarine_Spitfire.
aircraft fighter of RAF_Fighter_Command is Supermarine_Spitfire.

次に拡張情報として，「**aircraftFighter という関係を持つリソースの組**」を得る．この質問に対する SPARQL クエリは，

```
SELECT ?isValue ?hasValue WHERE { ?isValue
⟨http://dbpedia.org/ontology/aircraftFighter⟩ ?hasValue }
```

となる．

クエリはエンドポイントを通じて，結果，以下の知識（抽出されたものの一部を掲載）を得，ユーザに提示される．

aircraft fighter of United_States_Air_Force is F-16_Fighting_Falcon.
aircraft fighter of Turkish_Air_Force is F-16_Fighting_Falcon.
aircraft fighter of 53d_Test_and_Evaluation_Group is F-16_Fighting_Falcon.

`aircraft fighter of Air_National_Guard is F-16_Fighting_Falcon.`

（以上に示す実験結果は 2010 年 12 月 24 日現在の Wikipedia，DBpedia の資源に基づくものである.）

5. おわりに

本研究では，ユーザが興味をもった文章から，その文章の内容を最もよく表わしていると考えられる名詞を抜き出し（内容語と呼ぶ），内容語と強い関係を持つその他の名詞を関連語と読ぶ．そして，内容語と関連語の間にある関係を Linked Data を用いて抽出・提示することを提案した．また，そのようにして抽出した関係を持つ別のリソースを，Linked Data を用いて取得することにより，ユーザの興味に沿った情報拡張を行うことも提案し，その有用性を実データを用いて示した.

今後の課題として，現在の方法では選択された文章が著しく短かった場合，いかにして適切な内容語および関連語を見つけることが挙げられる．ユーザが興味を持つ文章が短文になることも多く，今後改良しなければならない問題である．また，現在は内容語と関連語の関係を見つけるのに，DBpedia の参照 RDF データに対して線形探索をしているにすぎない．そのため，内容語や関連語が RDF ファイルに含まれないような単語であった場合，有益な情報を取得することができない．今後はこのような場合でも，有益な情報を取得することが可能になるような手法に改良していくことが必要と考える．さらに，本研究で提案した手法の有用性の評価方法の検討と評価実験を行うことも重要であり，今後対処すべき課題として進めていくつもりである.

参 考 文 献

1) Berners-Lee, Semantic Web Road map(1998), http://www.w3.org/DesignIssues/Semantic.html
2) Berners-Lee, T.: Design Issues: Linked Data (2006), http://www.w3.org/DesignIssues/LinkedData.html
3) Bizer, C., Cyganiak, R., Heath, T.: How to publish Linked Data on theWeb (2007), http://www4.wiwiss.fu-berlin.de/bizer/pub/LinkedDataTutorial/
4) Geonames: http://www.geonames.org/
5) MusicBrainz: http://musicbrainz.org/
6) WordNet: http://wordnet.princeton.edu/
7) Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, Zachary Ives, "DBpedia: a nucleus for a web of open data", ISWC'07/ASWC'07: Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference, November 2007.
8) Li Ding, Tim Finin, Anupam Joshi, Rong Pan, R. Scott Cost, Yun Peng, Pavan Reddivari, Vishal Doshi, Joel Sachs, "Swoogle: a search and metadata engine for the semantic web", CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management, November 2004.
9) Watson: http://kmi-web05.open.ac.uk/WatsonWUI/
10) SWME: http://swse.deri.org/
11) Eyal Oren, Renaud Delbru, Michele Catasta, Richard Cyganiak, Holger Stenzhorn, Giovanni Tummarello, "Sindice.com: a document-oriented lookup index for open linked data", International Journal of Metadata, Semantics and Ontologies, Volume 3 Issue 1, November 2008.
12) Georgi Kobilarov, Tom Scott, Yves Raimond, Silver Oliver, Chris Sizemore, Michael Smethurst, Christian Bizer, Robert Lee, "Media Meets Semantic Web — How the BBC Uses DBpedia and Linked Data to Make Connections", ESWC 2009 Heraklion Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications, 2009.
13) Waitelonis, J. Sack, H., "Towards Exploratory Video Search Using Linked Data", 11th IEEE International Symposium on Multimedia, San Diego, CA, pp.540 - 545, 2009.
14) Yovisto: http://www.yovisto.com/
15) Bernhard Haslhofer, Elaheh Momeni, Manuel Gay, Rainer Simon, "Augmenting Europeana content with linked data resources ", I-SEMANTICS '10: Proceedings of the 6th International Conference on Semantic Systems, 2010.
16) Gunnar Aastrand, Remzi Celebi, Leo Sauermann, "Using linked open data to bootstrap corporate knowledge management in the OrganiK project", I-SEMANTICS '10 Proceedings of the 6th International Conference on Semantic Systems, 2010.
17) Christian Becker, Christian Bizer, "DBpedia Mobile: A Location-Enabled Linked Data Browser", 1st Workshop about Linked Data on the Web (LDOW2008), Beijing, China, 2008.
18) Alexander de Leόn, Victor Saquicela, Luis M. Vilches, Boris Villaz?n-Terrazas, Freddy Priyatna, Oscar Corcho, "Geographical linked data: a Spanish use case", -SEMANTICS '10: Proceedings of the 6th International Conference on Semantic Systems, September 2010.
19) SPARQL: http://www.w3.org/TR/rdf-sparql-query/