情報分析のための述語項構造を用いた動的オントロジー構築

村脇 有吾　　黒橋 禎夫
京都大学大学院情報学研究科
murawaki@nlp.kuee.kyoto-u.ac.jp kuro@i.kyoto-u.ac.jp

1 はじめに

Google をはじめとする既存の検索エンジンが検索結果として提示するのは、タイトル、URL、スニペットなど、ページごとの情報である。そのため、複数ページにまたがる知識を得るには、検索結果を 1 件ずつ見て回らなければならず、手間がかかる。従って、従来的なページごとの情報に代わる、新たなインターフェースが模索されている。

検索結果を組織化するユーザ・インターフェースの代表例として、クラスタリングと階層的なファセット分類が挙げられる [5]。クラスタリング検索エンジンの例としては、Vivisimo による Clusty[1] がある。ファセット分類としては、Stoica ら [6] が、検索結果のインターフェースとして、WordNet[4] などの静的な階層構造を圧縮したメタデータを提示する。

我々は、情報分析のための述語項構造を用いた動的オントロジー構築を提案する。このシステムは、検索エンジンが返す検索結果の文書群を解析し、文書群中の重要語についての述語項構造を整理して提示する。これにより、利用者が検索結果を俯瞰的に把握できるようにする。

また、従来の検索エンジンには、膨大な検索結果のうち、利用者が実際に閲覧するのは、上位のわずかなページに過ぎないという問題があった。これに対して、本システムを使うと、従来捨てられていた情報も有効に活用されるようになる。

2 概要

本研究が利用する述語項構造とは、動詞を介した名詞のペアの関係である。例えば、「グリーンピースは、一切の商業捕鯨に反対します。」という文は、サ変名詞 (動詞扱いする) の「反対」が、「グリーンピース」をガ格、「商業捕鯨」をニ格で支配しており、『反対 (グリーンピース:ガ，商業捕鯨:ニ)』という述語項構造が抽出される。同様にして、「捕鯨」に関する文書群を解析すると、『加盟 (捕鯨国:ガ，IWC:ニ)』、『行う (捕鯨船:ガ，捕鯨:ヲ)』、『再開 (ノルウェー:ガ，商業捕鯨:ヲ)』といった述語項構造が得られる。提案するシステムは、これらの述語項構造を整理し、図 1 のように提示する。単純な連想語データベースなどとは異なり、意味的に豊な述語項構造を用いることにより、利用者は、クエリについて、どのようなことが書かれているかをすばやく把握することができる。

国際捕鯨委員会 = IWC

- ～に 捕鯨国が 加盟
- ～が 捕鯨を 管理
- ～が 商業捕鯨を 中止

商業捕鯨

- ～に グリーンピースが 反対
- ～を ノルウェーが 再開
- ～を 国際捕鯨委員会が 中止

捕鯨

- ～を 捕鯨船が 行う
- ～に グリーンピースが 反対
- ～を 国際捕鯨委員会が 管理

図 1: 「捕鯨」の実行例 (抜粋)

3 手法

システムは、利用者からクエリを受け取ると、次の手順で、動的にオントロジーを構築する。

1. 検索結果から解析対象データを生成
2. 文書群から重要語を抽出
3. 重要語のうち、同義表現を併合
4. 重要語をグルーピング
5. 文書群から述語を介した重要語同士の関係を抽出

以下では、各ステップについて述べる。

3.1 解析対象データの生成

以下の手順で、検索エンジンからクエリに対応する文書群を取得し、解析対象となるデータを生成する。

1. 検索エンジンでクエリを引き、上位 n 件の文書を取得 (実験では $n = 400$)
2. 各文書から日本語文を抽出
3. 抽出された文の中から解析に用いる文を選択
4. 選択された各文がユニークか否かを検査

解析に用いる文を選択するのは、ブログのように、1 文書の中に、クエリとは無関係な話題が混在している文書への対策である。そのようなノイズを減らすために、検索キーワードを含む文とその周辺の文のみを解析に用いる。

次に、各文がユニークかを検査することにより、ドメイン名が異なる同一ページを解析から取り除く。同時に、ユニーク性の検査により、「リンク」、「コメント：」のように解析に不要な短い文字列もある程度排除できる。

実験では、検索エンジンとして、TSUBAKI[2] を用いた。TSUBAKI は、日本語ウェブ文書約 5,000 万ページを検索対象とする。検索対象文書へのアクセスに制限がないため、後述するように、全検索対象文書から、idf や同義表現のデータベースをあらかじめ構築できるという利点がある。

3.2 重要語の抽出

解析対象データから、以下の手順で重要語を抽出する。

1. 各文の形態素列から重要語候補を抽出
2. 重要語候補のスコアを計算
3. スコア上位 n 語を抽出 (実験では $n = 200$)
4. 重要語候補から不要な部分文字列を削除

重要語候補の抽出は、専門用語に複合名詞が多いことを考慮し、1 形態素だけではなく、複合名詞も対象とする。実験では、ストップワードなど、一部の例外を除き、連続する名詞性の形態素列のすべての組み合わせを重要語候補とする。例えば、「自然-言語-処理」という形態素列の場合、「自然」「言語」「処理」のほか、「言語処理」「自然言語処理」「自然言語」を抽出する。

抽出された重要語候補のスコア計算には、df-idf (document frequency-inverse document frequency) を用いる。

$$dfidf(t_i) = \sqrt{df(t_i)} \cdot \log idf(t_i) \quad (1)$$

$$df(t_i) = \frac{|d_i|}{|d_{total}|} \quad (2)$$

$$idf(t_i) = \frac{|D_{total}|}{|D_i|} \quad (3)$$

ここで、t_i はある重要語候補、$|d_{total}|$ は解析する文書の総数、$|d_i|$ は d_{total} のうち t_i を含む文書の数、$|D_{total}|$ はコーパス中の文書の総数、D_i は D_{total} のうち t_i を含む文書の数である。idf の値は、あらかじめ構築された idf データベースから取り出す。idf データベースは、上記の重要語候補抽出操作を、検索対象文書に適用して作成した。

この簡易的な手法では、不要な部分文字列も抽出される。例えば、「捕鯨」をクエリとしたとき、「国際捕鯨委員会」以外に「国際捕鯨委員」も残ってしまう。そこで、次のアルゴリズムで、不要な部分文字列を削除する。

1. 部分文字列の候補となる重要語候補のペアを抽出
2. それぞれのペアについて
   2.1 短い文字列を含む原文のうち、それが長い文字列の一部となっている割合を計算
   2.2 割合が閾値 (0.5) 以上の場合、短い文字列を破棄

3.3 同義表現の併合

抽出された重要語のうち、同義表現をまとめる。

同義表現には、「IWC」と「国際捕鯨委員会」に対する、「IWC総会」と「国際捕鯨委員会 総会」のような派生語も存在する。そこで、以下のアルゴリズムにより、重要語のペアが同義表現かを判定する。

1. 重要語のペアが同義表現データベースに含まれるなら真を返す
2. そうでなければ、先頭と末尾の共通する文字列を除去
3. 再び同義表現データベースを照合し、ペアが含まれるなら真、そうでないなら偽を返す

同義表現データベースは、idf データベースと同様に、検索エンジンが持つすべての文書からあらかじめ作成した。データベースの構築には、笹野らによる括弧表現を用いた同義表現獲得アルゴリズム [8] を利用した。

同義表現には、「狂牛病」、「BSE」、「牛海綿状脳症」のように 3 個以上の組み合わせもあり得る。そこで、

ある重要語が複数の同義表現ペアに含まれるとき、そのすべてを一つの同義表現の組とみなす。

なお、複数文書にまたがる解析に、単純な同義表現マッチングを利用することは、リスクを伴う。特に、頭字語などは、複数の語義を取り得るからである。例えば、IWC は、International Whaling Commission の頭文字であるとともに、International Watch Company の略でもある。しかし、今回のように、解析対象があるクエリを含む文書群である場合、ドメインが限定されるので、多義性を考慮しない単純マッチングでも問題はほとんどないと思われる。もちろん、「IWC」そのものをクエリとした場合、上記の 2 個 (あるいはそれ以上) の語義が混在してしまう。これはむしろ、あるクエリを含む文書群が複数のドメインにまたがるという、より一般的な問題である。例えば、「イースター」をクエリとする検索結果の文書群は、「復活祭」の他に「イースター島」に関するページも含まれる。従って、多義性の問題には、同義表現マッチングの改良よりも、あらかじめ解析対象の文書を分類し、ドメインを絞り込むことによって対処すべきと考える。

3.4 重要語のグルーピング

抽出した重要語には、「保護団体」、「環境保護団体」、「自然保護団体」あるいは「シロナガスクジラ」と「ニタリクジラ」のように、似たものが存在する。同義表現をまとめた時点で、重要語はスコア順に並んでいるが、利用の便を考えると、類似した重要語が近くに配置されることが望ましい。そこで、重要語同士の類似度に基づくクラスタリングにより、重要語を並べ替える。

類似度は、以下の表層的な手法で計算する。

1. 重要語のペアについて、共通するサフィックスのスコアを計算
2. 短い方の重要語の文字列全体のスコアで正規化

ここで、スコアは、字種のよって決まる 1 文字ごとスコアの総和である。1 文字ごとのスコアは、実験では、漢字が 3、カナが 1、記号とアルファベットが 0.5 とした。漢字の「病」は、カナの「ン」やアルファベットの「C」などよりも重要度が高いと考えられるからである。

この手法で、上記の「シロナガスクジラ」と「ニタリクジラ」などの組はクラスタ化される。一方、「捕鯨推進派」と「反対派」のように、語彙知識が必要な組はグルーピングされない。

類似度計算に、シソーラスのような知識体系を用いないのは、多くの複合名詞が静的な知識体系に収録されていないからである [3]。代わりに、修飾語が被修飾語の後に置かれるという日本語の性質を考慮して、共通するサフィックスを用いている。未定義語への対策としては、他に、シソーラス照会に複合名詞の末尾の形態素を使うという手法も考えられる。あるいは、既存の階層的な知識体系に未定義語を動的に追加するという方法も考えられる。

3.5 述語項構造の抽出

述語項構造の抽出には、ウェブから自動獲得した大規模格フレーム [7] を用いる。格フレームを用いることによって、連体修飾や、ハ、モのような係助詞を使うため明示されない格の推定が行える。

述語項構造を抽出する際、原則的に付属語を取り除いて抽象化する。このため、付属語によって表現されるテンス、アスペクト、モダリティなどの情報は落とされる。ただし、否定表現は、取り除くと意味が反対になってしまうので取り除かない。否定表現を含む場合、述語項構造に否定のラベルをつける。受身と使役も、動詞が支配する格が交替するので、識別が必要である。今回用いた格フレームが、受身と使役を通常の動詞と区別しているので、その情報を利用する。

文書群から抽出される述語項構造のうち、重要語のペアを格スロットに持つものを抽出する。ただし、格スロットに入る複合名詞と重要語が完全一致するとは限らないので、重要語が格スロットの末尾に含まれるとき、重要語のペアを格スロットに持つと判定する。

最終的な出力は、重要語のリストを見出しとし、抽出された述語項構造は、該当する各重要語の下に配置する。また、述語項構造をクリックすると抽出元の原文が表示されるようにする。

4 考察

問題の性質上、定量的な評価を行うことが難しい。ここでは、クエリ「狂牛病」の実行結果について、検索エンジンの出力と比較して考察する。

検索エンジンが返す文書には、新聞記事や新聞記事へのリンク集、あるいは狂牛病に関する個々の事例などが多い。狂牛病に関する情報を把握するには、複数の文書を読みくらべて自分で情報をまとめるか、リンクをたどって簡潔にまとめられたページを見つけなければならない。一方、図 2 に示す本システムの出力を眺めると、狂牛病に関する基本事項をおおよそ把握することができる。これは、予備的な実験としては良い結果と考える。

狂牛病 = 牛海綿状脳症 = BSE

- ～と 牛を 診断
- ～と 牛を 判断
- ～に 牛が 居る
- ～に 牛が 掛かる
- ～に 牛が 感染
- ～を 牛が 発症
- ⋮
- ～で 牛が 死ぬ
- ～に 乳牛が 感染
- ～が 英国で 発生

牛

- ～を 英国から 輸入
- ～を 狂牛病と 診断
- ～を 狂牛病と 判断
- ⋮
- ～が 狂牛病で 死ぬ
- ～について 検査を 実施
- ～を 検査が 実施
- ～が 検査を 仕舞える
- ～に 肉骨粉を 与える
- ～が 肉骨粉を 食べる

肉骨粉

- ～を 英国から 輸入
- ～を 飼料が 含む
- ～を 飼料として 与える
- ～を 牛に 与える
- ～を 牛が 食べる

図 2: 「狂牛病」の実行結果 (抜粋)

逆に、うまくいかない例に「アルゴリズム」がある。「アルゴリズム」をクエリとして得られる文書群に、アルゴリズムそのものを説明したページは少ない。むしろ、暗号のアルゴリズムや遺伝的アルゴリズムといった、相互にあまり関連のない具体的なアルゴリズムに関する説明が大半を占める。このため、解析結果は、内容に一貫性を欠いてしまっている。これは、検索エンジンが本当に利用者が求める結果を返しているのかという本質的な問題にかかわるもので、今後の課題とする。

抽出する述語項構造を充実させるには、照応・省略解析が必要となる。例えば、

> タネンバウムはアメリカ合衆国のニューヨーク市で生まれ、ニューヨーク州のホワイトプレインズ市で育った。

という文について考えたとき、「タネンバウムは」の係り先は「育った」なので、『育つ (タネンバウム:ガ, ホワイトプレインズ市:デ)』が抽出される。一方、「生まれ」には「タネンバウムは」が係らないので、抽出される述語項構造は『生まれる (ニューヨーク市:デ)』となる。『生まれる (タネンバウム:ガ, ニューヨーク市:デ)』という述語項構造を生成するには、『生まれる』のガ格が「タネンバウム」であることを推定しなければならない。現時点の自然言語処理の技術では、本格的な照応・省略解析結果の利用は精度上問題があるかもしれない。しかし、この例のように、信頼性の高そうな解析結果のみを利用できないかと検討している。

5 おわりに

述語項構造を用いて動的にオントロジーを構築することにより、利用者に検索結果の俯瞰的な把握を可能とするシステムについて述べた。

このシステムの実用化には、処理速度が問題となる。静的なデータの利用による動的処理の削減や、処理の並列化といった工夫が欠かせない。

述語項構造を使った知識表現は、概念同士が有機的なつながりを持っているため、高度な推論への応用が考えられる。しかし、そのためには、技術的な課題が少なくない。上述の照応・省略解析以外にも、表現のずれを吸収する仕組みは不可欠である。例えば、牛が狂牛病に感染したことを説明する場合、「牛が狂牛病にかかる」、「牛が狂牛病になる」、「狂牛病が牛に発生する」、「牛が狂牛病を発症する」というように、多様な表現がなされる。これらが同一の事象であることを計算機に認識させる必要がある。

参考文献

[1] Clusty the clustering search engine. `http://clusty.com/`.

[2] 検索エンジン基盤 TSUBAKI. `http://tsubaki.ixnlp.nii.ac.jp/se/index.cgi`.

[3] M. Ciaramita and M. Johnson. Supersense tagging of unknown nouns in wordnet. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pp. 168–175, 2003.

[4] C. Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[5] M. A. Hearst. Clustering versus faceted categories for information exploration. *Commun. ACM*, Vol. 49, No. 4, pp. 59–61, 2006.

[6] E. Stoica and M. A. Hearst. Nearly-automated metadata hierarchy creation. In *HLT-NAACL: Short Papers*, pp. 117–120, 2004.

[7] 河原大輔, 黒橋禎夫. Web から獲得した大規模格フレームに基づく構文・格解析の統合的確率モデル. 言語処理学会 第 12 回年次大会 発表論文集, pp. 1111–1114, 2006.

[8] 笹野遼平, 河原大輔, 黒橋禎夫. 自動獲得した知識に基づく統合的な照応解析. 言語処理学会 第 12 回年次大会 発表論文集, pp. 480–483, 2006.