Web ページ検索結果の絞込みのための記述要素の提示

久保木 武承, 山本 和英
長岡技術科学大学 電気系
E-mail:{kuboki, yamamoto}@jnlp.org

1　はじめに

検索エンジンは Web で情報を取得する時、たびたび利用される。しかし検索エンジンによりユーザの望んだページを探すことは難しい。ページを見つけ出すためにユーザはキーワードを工夫し、何度も検索を行わなくてはならない。

その原因の一つに、キーワードによる検索はページ本文の内容を考慮していないことが挙げられる。通常、検索結果はキーワードが含まれているページを提示しているだけであり、キーワードがそのページにとってどのような意味を持っているかを考慮していない。このため検索エンジンはページ中の文章がどのような事を書いているかにかかわらず検索結果を提示してしまう。

この問題への既存の手法による対処としては、スニペットの利用などが挙げられる。スニペットとはキーワード周辺の文章を抜粋したもので、本文の内容を端的に示す目的で使われている。しかしスニペットはあくまで抜粋であり、本文の内容を考慮して提示しているわけではない。

検索結果として提示されるページのタイトルも、検索においては重要な要素である。しかし Web 検索で表示されるタイトルとは、Web ページの作成者がそれぞれ自由につけた物であり、実際にはページの内容を正確に表現していない事が多々ある。従って、タイトルをそのまま使用して検索にフィルタをかけるといった処理は、タイトルの正確さが不明であるため有効ではない。

だが実際の検索では、ユーザがページの内容を判断するにあたり、タイトルやスニペットは重要な要素となっている。機械的な処理ではなく、ユーザが人手で見る限りは、そこからページの内容を類推することができるためである。

けれどそれはあくまで類推であり、タイトルやスニペットを用いてもページ中の文章がどのような目的で書かれた物なのか、例えば概要を書いたのか、紹介を書いたのか、歴史を書いたのか、産地に関する説明を書いたのか、といった事を察することは難しい。結果として、中国の歴史を調べていたのに、中国の歴史を書いた本を紹介するページしか見つけられないといった事がある。

本稿では、この問題を解決するには検索の際にユーザにページ中の本文が「何の目的で書かれた」かを提示する事を提案する。

ページ中の本文が「何の目的で書かれたか」を表現するにあたり、本稿では Wikipedia に見られるような節見出しに着目した。節見出しは「概要」「歴史」「出典」「関連商品」といった短いタイトルをつけることで、その先の本文が何の目的で書かれたかを簡潔に表現している。

タイトルやスニペットに付随して、ページの本文が何の目的で書かれたのかという文書の記述要素を表示することで、ユーザはページ本文を一々調べることなく、そのページ中にどのようなことが書かれているか即座に把握することができる。くわえてこの手法なら、スニペットのようなキーワード周辺の抜粋とは異なりより普遍的な本文の特徴をユーザに提示する事が可能である。

そこで本稿では、Web 検索の際にユーザがより容易にページの中身を理解するために、本文の特徴を示す記述要素の生成を行った。

関連研究する研究として、Web 検索結果の選択を支援するため、検索結果をジャンルで分類する研究がある。伊藤らは Web 検索サービスでの検索結果を事前に設定した 8 種類のジャンルとその下に据えた子ジャンルに分類し、従来の検索結果に付加する形で提示するシステムを提案した [1]。

記述要素により Web 上の文書を分類する手法は、検索用ディレクトリに近いとも言える [2]。検索用ディレクトリはあらかじめキーワードにより Web ページが分類されている状態にある。

2　提案手法

2.1　全体像

システムの流れを以下に示す。

1. ユーザによるクエリの入力
2. 検索結果の取得
3. 検索結果の Web ページの本文から記述要素を生成
4. 検索結果と生成した記述要素を提示

本稿では上記の流れに沿った検索システムの流れを想定し、図中における「検索結果の Web ページの本文から記述要素を生成」するシステムの設計と実験を行った。

以下で提案する検索システムの詳細から記述要素の生成方法まで、順を追って説明する。なお、記述要素の定義は 2.3 節で行った。

2.2　検索システムの提案

本稿で提案する検索システムでは、事前にコーパスを用いて作成した記述要素の特徴と Web 検索で得られたページ本文の特徴を比較し、特徴が合致した記述要素をページ本文の記述要素として提示している。

検索システムの流れを図 1 に示す。

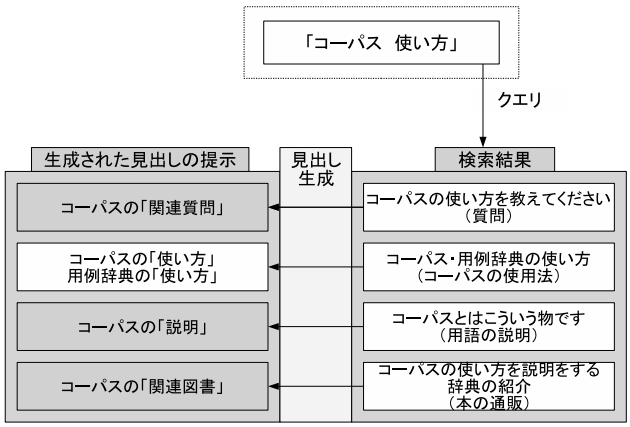

図 1: 検索システムの概念図

図 1 ではまず通常の Web 検索と同様に、キーワードとして「コーパス　使い方」の 2 語をユーザが選んで検索を行っている。

しかしこのままではページの内容を考慮することができず、使い方の説明文を探している時に質問のページや通販のページ、または用語の説明を行うページが同時に提示されてしまう。

通常、ユーザは検索結果として提示されたタイトルやスニペットからページの内容を類推するか、直にページ本文を見て、そのページの本文がどのような目的で書かれたかを判断をしなくてはならない。

しかし類推することは検索に慣れていないユーザや不慣れな分野を検索するユーザにとっては難しく、直にページ本文を見る

にしても大量の検索結果が提示されては全てを見るのは大変である。

この問題は、そのページの本文がどのような目的で書かれたかはタイトルやスニペットだけでは判断できないことから発生している。タイトルやスニペットは図 1 の左部分のように、本文が何の目的で書かれているか、例えば何かの関連質問を説明するために書かれたのか、何かの使い方を説明するために書かれたのか、何かの関連図書を説明するために書かれたのかといった事を表現しているとは限らないたのである。

この問題の解決を目指して、本稿では検索結果が得られた後に図 1 の左部分の「」で囲まれてるような記述要素を生成するシステムを提案する。

このような記述要素をタイトルやスニペットと共に提示する事により、ユーザはページ本文を一々確認する事無くそのページの本文がどのような目的で書かれたかを把握する事が可能となる。

ここで作り出す記述要素とは、本文が何の目的で書かれたかを説明するものである。すなわち、図 1 の左部分の「」で囲まれてるような記述要素を提示する事で、ユーザがより容易に検索を行う事を可能とする。

2.3 命題と記述要素

記述要素の定義にあたり、「本文が何の目的で書かれているか」という事をより明確にする必要がある。そこで本稿では、検索に用いる記述要素の生成を目的としていることを受けて、クエリの存在、より正確にはユーザが抱く質問から記述要素の定義を検討した。

本稿では「本文が何の目的で書かれているか」を説明する記述要素を作るため、「〜とは何ですか」という What 型の質問をモデルとした。藤岡らは What 型質問は「名詞でなく文章で回答するシステムが必要とされる」と書いており、この定義は上記記述要素の定義に合致する [3]。

さらにここでは、記述要素の定義のために What 型の質問文を以下のような構造として取り扱った。

・［命題］の＜記述要素＞は？

この例としては、以下のような物がある。

・［コーパス］の＜使い方＞は？
・［りんご］の＜産地＞は？
・［ローパスフィルタ］の＜機能＞は？

このように、本システムが生成すべき記述要素とは、質問から命題を除いた物、つまり「……の概要」「……の歴史」「……の機能」といった、本文で説明する内容を端的に表した言葉と定義する。

また命題とは「コーパス」のような検索の主な対象となる語として定義し、「［命題］の＜記述要素＞は？」の形式に各々の語を当てはめた時、可読性の高い物とする。

2.4 記述要素の特徴データに用いるコーパス

本システムでは、記述要素は 2.3 節の定義に沿って事前に用意しておき、入力された本文に対してどれが適当か選択する。そのため、事前に大量の記述要素と本文の組み合わせを取得しておく必要がある。

そのためのコーパスとして、本稿では Wikipedia の節見出しと本文の関係に着目した (図 2)。

節見出しは Wikipedia の各項目の目次に記載されている物の事で、これは本文の意味を説明しているという点で、2.3 節の定義に沿っている。

そして Wikipedia には多岐にわたる命題の説明文が存在しており、その多くが節見出しによって分類されている。従って Wikipedia をコーパスとして用いる事で大量の節見出しと本文のセットが得られる。

また、Wikipedia からは異なる命題で同じ節見出しを抽出することができる。これを利用し、複数の命題にわたり共通の節見出しの本文を得る事で、命題の影響を無視して本文の傾向を得ることができる。

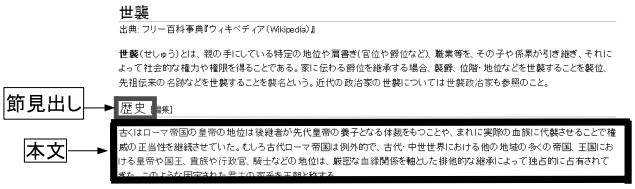

図 2: 節見出し

2.5 記述要素の特徴データ

記述要素の特徴データの生成のため、最初に Wikipedia から節見出しと本文のセットを取得する。次に本文中から、共通する記述要素の特徴データを得る。

本手法では本文の特徴データとして、形態素、および形態素の概念の tfidf を取得した。なお、tfidf の計算の際には、1 記事を「節見出しと本文のセット」として扱った。

また、上記の方法で特徴データを得るために、Wikipedia 上に複数存在する節見出しを取り扱わなければならない。そこで Wikipedia 上に 3 つ以上存在する節見出しを取り扱った。

記述要素の特徴データとして用いるのは本文中の動詞、名詞、形容詞とした。これらの品詞に限定した理由は、本文中でも特に強い特徴となると考えたためである。

他、これらの語の概念も取得して、別個の特徴として使用した。

ただし語の概念とは、利用する語を直接示す概念 (0 レベルのノードの概念) のみを用い、それより遡っての取得 (1 レベル以上のノードの概念) はしていない。また 1 つの語につき複数の概念を得られた場合、その全てを利用した。

これらの要素 (形態素、概念) に対して、tfidf 値を求めた。

なお、形態素解析機は茶筌 (2) を、概念の取得には EDR 電子化辞書 (4) の概念辞書を用いた。

3 実験

3.1 入力データの前処理

正解候補の出力にあたり、Web ページにある本文が入力となる。

入力された本文は 2.5 節と同様に動詞、名詞、形容詞だけを抽出し、形態素、概念でまとめた。これを入力データの特徴として使用した。

3.2 正解候補の選定

2.5 節で生成した記述要素の特徴データと 3.1 節で生成した本文の特徴データを比較し、本文に最も近い記述要素の特徴データを選定する。

本稿では以下の 2 種類の方法でスコアを付与し、最もスコアの高いものを正解候補として提示した。

手法 1　入力した本文の特徴と一致した要素の数が最も多い記述要素を正解候補とする
手法 2　正解候補の要素に順位に従った重み付けを行い、手法 1 と同様に存在する要素を選び出し、合計値が最も高い記述要素を正解候補とする

手法 2 で用いた重み付けは、上位 n 件の要素と一致させる時、「n/要素の順位」とした。なお要素の順位とは、記述要素のもつ要素を tfidf 値の高い順で整理した時の順位である。

ただし、上記の手法で用いる要素の数は以下の 3 つにわけた。これは要素数が記述要素の選定に及ぼす影響を調べるためである。

・記述要素の特徴データの tfidf 値上位 100 件の要素
・記述要素の特徴データの tfidf 値上位 1000 件の要素
・記述要素の特徴データの tfidf 値上位 10000 件の要素

ただし記述要素の特徴データの要素数がそのときの必要とする要素の数 (100 件、1000 件、10000 件) より提示した物よりも少なかった場合、その全てを使うものとする。

3.3 実験方法

実験ではクローズドテスト、オープンテストをそれぞれ行った。実験は 3.2 節に示した方法で正解候補を出力し、再現率、適合率、および F 値を求めた。

3.4 クローズドテスト

クローズドテストでは、Wikipedia 中に存在する互いに異なる記述要素 141392 件からランダムに 100 件選び、その要素を入力した。ただし記述要素は 2.5 節の条件と同じ、Wikipedia に 3 件以上存在するものとした。

3.5 オープンテスト

オープンテストでは、クローズドテストと同様の記述要素について、作業者 1 人が Google で検索して入手した本文をテストセットとして使用した。ただしテストセットの作成時には 2.3 節で定義した命題に関しては考慮せず、ページ本文が記述要素と合致していると判断した物を採用した。

4 実験結果

4.1 Wikipedia を用いた記述要素特徴データの生成

Wikipedia から得られた記述要素の件数を表 1 に、見出しごとの要素数に関するデータを表 2 に示す。

表 1: Wikipedia から得られた記述要素データ

見出しの数 (重複あり)	433473
見出しの数 (重複無し)	141392
3 個以上重複している記述要素の数	8436
2 個重複している記述要素の数	7461
1 つしかない記述要素の数	125495

表 2: 使用する記述要素ごとの要素数

種類	要素数の最大値	要素数の最小値	要素数の平均値
形態素	50422	2	463
概念	39534	1	614

表 1 より、本実験で用いる記述要素は 8436 件となる。

表 2 はこの 8436 件のデータに対して調査した結果であり、Wikipedia に 2 個以下しか存在しない記述要素については考慮していない。ただし、3 件以上存在する記述要素についてもその要素数は安定しておらず、形態素、概念ともに最大値と最小値の間に開きが見られた。

4.2 クローズドテスト／オープンテスト

表 3、表 4、表 5、表 6 に実験結果を示す。

クローズドテストで最も高い評価を得たのは手法 1、手法 2、共に形態素を要素とした物で、F 値は 0.89 だった。概念を用いた時の最大の F 値は、手法 1、手法 2 共に 0.8 だった。この全ての条件において、要素数は上位 10000 件を用いた時のものであった。

一方オープンテストでは、F 値は最大で 0.04 と低い値になった。このスコアを出したのは手法 1 の形態素、概念と、手法 2 の概念によるシステムである。用いた要素数はいずれも上位 1000 件のものだった。

クローズドテストで高い F 値を得る事ができた上位 10000 件の要素を用いた手法は、オープンテストではいずれも上位 1000 件の要素を用いた方法に比べて低いスコアを出した。

表 3: 手法 1(形態素) の結果

	再現率		適合率		F 値	
要素	closed test	open test	closed test	open test	closed test	open test
上位 100	0.36	0.03	0.34	0.03	0.35	0.03
上位 1000	0.82	0.04	0.75	0.04	0.78	0.04
上位 10000	0.96	0.02	0.83	0.02	0.89	0.02

表 4: 手法 1(概念) の結果

	再現率		適合率		F 値	
要素	closed test	open test	closed test	open test	closed test	open test
上位 100	0.21	0.01	0.18	0.01	0.19	0.01
上位 1000	0.73	0.04	0.62	0.04	0.67	0.04
上位 10000	0.94	0.03	0.69	0.03	0.80	0.03

表 5: 手法 2(形態素) の結果

	再現率		適合率		F 値	
要素	closed test	open test	closed test	open test	closed test	open test
上位 100	0.19	0.00	0.18	0.00	0.18	0.00
上位 1000	0.73	0.03	0.66	0.03	0.69	0.03
上位 10000	0.96	0.02	0.83	0.02	0.89	0.02

表 6: 手法 2(概念) の結果

	再現率		適合率		F 値	
要素	closed test	open test	closed test	open test	closed test	open test
上位 100	0.12	0.00	0.10	0.00	0.11	0.00
上位 1000	0.68	0.04	0.57	0.04	0.62	0.04
上位 10000	0.94	0.03	0.69	0.03	0.80	0.03

5 考察

5.1 オープンテスト、クローズドテストの結果比較

手法 1、手法 2 どちらにおいても、クローズドテストに比べてオープンテストの評価は下がった。この理由は 4.2 節の表より、記述要素の特徴データで用いた要素と入力データの要素の差異による物である。

4.2 節の実験結果では、クローズドテストは使用する要素の数を増やすほど評価は平均して向上した。クローズドテストでは、入力に用いる全ての要素は正解となる記述要素の特徴データに含まれている。

このとき、クローズドテストで再現率が 1.00 にならない理由は以下の通りである。

本実験では記述要素の特徴データのうち、tfidf 値の高い要素上位 100 件、1000 件、10000 件を使用した。しかしその結果、入力された要素が正解の記述要素の特徴データに存在しないという事が起こり、他に要素の存在する記述要素を正解候補として出力した。

要素の使用する数を増やすほど評価が向上したのも同様の理由によるものである。

一方、オープンテストではクローズドテストとは異なり、入力される要素が正解である記述要素の特徴データに存在しない可能性がある。本実験ではこの対策として、記述要素の正解データを作る時に各要素の頻度情報から tfidf 値を求めて用いたが、結果を向上させることにはつながらなかった。

5.2 記述要素の特徴データ

5.1 より、オープンテストの結果で F 値が低かった理由は見出しの特徴データにあるとわかった。そこで入力データと正解となる記述要素の特徴データ間の要素が一致した割合を確認した。

正解となる記述要素の特徴データの全ての要素で確認した結

果を表 7 に、見出しの特徴データの上位 1000 件の要素で確認した結果を表 8 に示す。

なお、ここで求めた値を式で表すと、

$$\frac{\text{入力した本文の要素が記述要素候補の要素と一致した数}}{\text{記述要素の特徴データの要素数}}$$

となる。

表 7: 要素の一致割合 (全要素)

	形態素		概念	
データの範囲	closed test	open test	closed test	open test
0.00-0.05	38	30	46	29
0.05-0.10	29	27	21	29
0.10-0.15	12	15	16	15
0.15-0.20	8	9	5	10
0.20-0.25	1	9	2	5
0.25-0.30	2	4	2	4
0.30-0.35	1	2	2	3
0.35-0.40	4	2	1	2
0.40-0.45	0	1	1	3
0.45-0.50	1	1	1	0
0.50-0.55	1	0	0	0
0.55-0.60	0	0	1	0
0.60-0.65	1	0	1	0
0.65-0.70	0	0	0	0
0.70-0.75	1	0	0	0
0.75-0.80	0	0	0	0
0.80-0.85	0	0	0	0
0.85-0.90	0	0	1	0
0.90-0.95	0	0	0	0
0.95-1.00	1	0	0	0

表 8: 要素の一致割合 (上位 1000 要素)

	形態素		概念	
データの範囲	closed test	open test	closed test	open test
0.00-0.05	38	27	45	27
0.05-0.10	29	31	21	31
0.10-0.15	12	15	17	15
0.15-0.20	8	8	5	11
0.20-0.25	1	9	0	5
0.25-0.30	2	5	3	3
0.30-0.35	1	1	3	4
0.35-0.40	4	2	1	1
0.40-0.45	0	1	1	3
0.45-0.50	1	1	1	0
0.50-0.55	1	0	0	0
0.55-0.60	0	0	1	0
0.60-0.65	1	0	1	0
0.65-0.70	0	0	0	0
0.70-0.75	1	0	0	0
0.75-0.80	0	0	0	0
0.80-0.85	0	0	0	0
0.85-0.90	0	0	1	0
0.90-0.95	0	0	0	0
0.95-1.00	1	0	0	0

表 7 より、クローズドテストとオープンテストの間で、正解となる記述要素の特徴データとの要素の一致数に大きな差は見られない事が分かった。これは形態素を用いた手法、概念を用いた手法どちらでも変化はなく、いずれも 5%から 0.15%の間に分布している。この傾向は表 8 でも同様である。

このことから、記述要素の要素の一致する割合はクローズドテストとオープンテストの結果に差を作る原因ではなかったと考えられる。

そこで次にオープンテストとクローズドテストのテストセットに存在する要素の数を検討する。表 9 に記述要素ごとにクローズドテストの要素の数からオープンンテストの要素の数の差を求め、平均した結果を示す。

表 9: 要素の一致割合 (上位 1000 要素)

種類	平均要素数差分値
形態素	-212
概念	-299

表 9 より、平均してオープンテストの持つ要素の数はクローズドテストの要素の数よりも大きい事がわかる。その差は形態素では 212 個であり、概念では 299 個である。

この影響を受けて、オープンテストでは正解とは異なる記述要素の要素を多数含み、結果として不正解の記述要素を選んだ。

その対応として手法 2 で要素の重み付けを行う実験を行ったが、手法 1 との間に結果の違いは見られなかった。このことより、tfidf による要素への重み付けもオープンテストのテストセットには有効では無かったといえる。

6 おわりに

本稿では、検索に用いるための本文から記述要素を生成する手法を提案した。処理には Wikipedia を用い、記述要素と本文の組み合わせを取得して、Web から得られた本文と一致する特徴を持つ記述要素を正解候補として出力した。しかし実験の結果、クローズドテストでは F 値で 0.89 の値を出すことができたが、オープンテストで得られた F 値は最大で 0.04 と低い値になった。

その原因は記述要素の特徴データとオープンテストのデータの違いにある事が分かった。オープンテストは正解となる記述要素の特徴データとは異なる要素を多分に含んでいた。

今後の課題としてはオープンテストでの精度の向上が挙げられる。

そのための対策としては、一つに記述要素の特徴データをよりオープンテストに近い環境、すなわち一般的な Web テキストから作り出す事が考えられる。また、記述要素の特徴データに共起等の頻度以外の情報を用いる事も検討する必要がある。

他方では、本実験では入力データの要素が多い事が問題点としてあげられている。これを解決するため、入力データから得られる要素を何らかの方法で選別する、といった方法も考えられる。

使用した言語資源及びツール

(1) フリー百科事典　ウィキペディア日本語版, 20090124 時点
http://ja.wikipedia.org/wiki/

(2) 形態素解析器「茶筌」,Ver.2.3.3, 奈良先端科学技術大学院大学 松本研究室,
http://chasen.naist.jp/hiki/ChaSen/

(3) IPA 品詞体系日本語辞書「IPADIC」, Ver.2.7.0, 奈良先端科学技術大学院大学 松本研究室,
http://chasen.naist.jp/stable/ipadic/

(4) EDR 概念辞書, 情報通信研究機構 (NICT)
http://www2.nict.go.jp/r/r312/EDR/

参考文献

[1] 伊東敏章, 池田尚志. Ｗｅｂ検索結果のページ選択を支援するジャンル分類システム. 言語処理学会第 14 回年次大会, pp.955-958,2008.

[2] 隅田飛鳥, 後河内脩平, 三浦二三高, 相川昌裕, 鳥澤健太郎. WWW 文書集合から自動抽出した意味的関係を用いた大規模な検索用ディレクトリの試作. 言語処理学会第 13 回年次大会, pp.1121-1124,2007.

[3] 藤岡秀明, 浦谷則好. What 型Ｑ＆Ａシステムの構築. 言語処理学会第 15 回年次大会, pp.52-55,2009.