検索エンジンを用いた関連語検索システムの設計と実現

安藤大幸† 寺島浩太† 藤本敬介† 中山泰一†
†電気通信大学 情報工学科

1 はじめに

我々は用語の意味を知る際に，用語そのものの意味を知ろうとする．しかしそれだけではなく，用語の関連語やそれらの関係を知ることが，元の用語の意味を更に理解する手助けになる．そして，このような用語の関連についての知識から，文書同士の比較や情報検索における利便性の向上が期待できる．

そこで本研究では，入力されたクエリを元に，Web からそれに関連すると思われるキーワードを抽出し，キーワードの関連度を定量化するシステムの構築と評価を行う．

2 関連研究

関連語を検索する研究として，芳鐘ら [1] は「生体計測」「生体の計測」というような言い換え表現を考慮した上で，修飾関係にある語から関連語を収集する手法を提案している．

また，関連語の応用として，検索クエリを提示するシステムがある．その中で代表的な Google Suggest[2] は，Google 利用者が入力したクエリの中から，人気のあるクエリを提示してくれるというものである．他にも，野田ら [3] は，「A の B」というような「の」で繋がる語から，絞り込み検索に有効なキーワードを発見するという手法を提案した．

しかし，これらはあくまで検索クエリの提示を目的とするものであり，今回の研究では関連語の全般の収集を行うものとする．

3 システムの設計

3.1 関連語検索の流れ

関連語を検索するための元となるクエリ q を検索クエリとして，検索エンジンを使って Web 検索をする．そして得られる Web ページのアドレスを元に HTML をダウンロードし，HTML タグなどを除去して文章群 $S(q)$ を取得する．$S(q)$ を形態素解析し，得られた名詞群 $N(q)$ に対して関連度の計算を行うことで，最終的な関連語の候補を抽出する．

3.2 関連度の計算

クエリ q を元に得られた名詞群 $N(q)$ に対し，関連度が高いものは出現頻度が高くなるという考え方から，まず出現頻度が高い上位の何件かを選び出す．しかし，頻度情報だけを元にして関連語を抽出すると，関連性の高いキーワードと共に，例えば「大学」や「研究」といった，一般性の高いキーワードも選び出されてしまう．

そこで，寺島 [6] が考案した「依存度」という概念を用いて，関連性が低い単語の除外を行う．検索エンジンから得られるクエリ x のヒット数を $df(x)$ として，クエリ q に対する名詞 $n \in N(q)$ の依存度 $dep(q,n)$ を，n と q を用いて以下のように定義する．

$$dep(q,n) = \frac{df(n \text{ AND } q)}{df(n)}$$

$dep(q,n)$ は，n と q の AND 検索ヒット数を n の検索ヒット数で割ったもの，つまり Web において n が出現するドキュメントのうち，q が含まれるドキュメントの割合である．

依存度のみを用いて関連度を計算すると，固有名詞のような特徴的なキーワードばかりが選び出されてしまう．例えば「Python」という単語に対して「Perl」や「Ruby」という単語が抽出された場合，ユーザはこれらの単語に関連があることは分かっても，「Python」や「Perl」が具体的に何を示しているのかは分からない．そのため，このような場合は「スクリプト」や「プログラム」といった一般名詞も抽出されるべきである．そこで，上記の「関連度が高いものは出現頻度が高くなる」という考え方を考慮し，n の出現頻度を $freq(n)$ として，q に対する n の関連度 $rel(q,n)$ を定義すると以下のようなる．

$$rel(q,n) = dep(q,n) \times freq(n)$$

Search system for related words using Web Search Engines
Hiroyuki ANDO†, Terashima KOTA†, Fujimoto KESUKE†and Nakayama YASUICHI†
†Department of Computer Science, The University of Electro-Communications
182-8585, Chofu, Japan

3.3 実装

入力キーワードの検索には Yahoo! JAPAN が提供する Yahoo!検索 API[4] を，形態素解析には奈良先端科学技術大学院大学松本研究室が開発する MeCab[5] を，システム全体の実装には Perl を，それぞれ用いた．

4 実験結果と考察

クエリを「python」に設定した場合の抽出結果上位 15 件を表 1 に示す．

表 1: q="python" の抽出結果

単語	$freq(q)$	$dep(q,n)$	$rel(q,n)$
Jython	16	0.717352	11.477639
SWIG	19	0.596000	11.324000
Perl	34	0.131066	4.456236
スクリプト言語	17	0.153846	2.615385
プログラミング言語	25	0.095122	2.378049
GUI	19	0.102994	1.956886
Java	28	0.062876	1.760515
オブジェクト	16	0.060641	0.970262
ライブラリ	31	0.025841	0.801062
言語	37	0.020830	0.770705
sample	42	0.016647	0.699162
スクリプト	16	0.039258	0.628133
開発	25	0.007547	0.188668
win 32	13	0.010829	0.140777
Web	21	0.005438	0.114188

表 1 を見ると，上位の単語は「Jython」や「SWIG」など，「python」に関連したものが抽出されているといえる．また，「スクリプト言語」と「言語」を比較すると，出現頻度は「言語」が多いものの，依存度の計算により，最終的に「スクリプト言語」の方が上位になっている．一般性の高い「言語」という単語よりも「スクリプト言語」の方が関連度が高いという結果は，好ましいものであると考えられる．

表 2: q="mac" の抽出結果（抜粋）

単語	$freq(q)$	$dep(q,n)$	$rel(q,n)$
アップル	136	0.276470	37.600000
Leopard	19	0.822727	15.631818

また，表 2 を見ると，「mac」に対する依存度は「Leopard」の方が高いが，最終的には一般性の高い「アップル」の方の関連度が高くなっている．依存度だけで関連度を決定していたら「アップル」は抽出されないほど関連度が低くなってしまうが，出現頻度との積をとっているため，一般性の高いキーワードも抽出することが可能である．

5 おわりに

本研究では，依存度と出現頻度から，関連語の検索システムの構築を目標とする実験を行った．実際に検索を行ったところ，関連語自体の抽出は比較的期待通り動作を示した．しかし，ノイズとなる単語が抽出されることも多く，どのような単語に対してでも確実に関連語を抽出できるわけではなかった．

また，依存度によるソート結果と関連度によるソート結果を比較したところ，多少の違いが見られたものの，ほとんど同じ結果になることが判明した．これは，関連度に対する依存度の占める割合が大きく，出現頻度の変化があまり反映されていないことに起因すると考えられる．

そして，抽出された単語が関連語かどうかの判断は主観によるものなので，実際にどれくらい関連している単語なのか，定量的な評価が必要であると考えられる．

参考文献

[1] 芳鐘 冬樹，野澤 孝之，辻 慶太，影浦 峡：ウェブからの関連語・下位語の収集手法の検討と検索システムへの応用，第 52 回日本図書館情報学会研究大会発表要綱，pp.113-116.

[2] Google Suggest
http://www.google.co.jp/webhp?complete=1&hl=ja

[3] 野田武史，大島裕明，小山聡，田島敬史，田中克己：主題語からの話題語自動抽出とこれに基づく Web 情報検索，日本データベース学会 Letters Vol.5, No.2, pp.69-72(2006)

[4] Yahoo!デベロッパーネットワーク
http://developer.yahoo.co.jp/

[5] MeCab
http://mecab.sourceforge.net/

[6] 寺島浩太，安藤大幸，藤本敬介，中山泰一：Web 検索のための有用な関連キーワードを評価するシステムの構築，第 70 回情報処理学会全国大会