

緩和度付き検索語の意味関連分析による検索意図推定とそのクエリ入力インタフェース

Detecting Search Intention by Analyzing Relationship between Keywords with Relaxation Value and an Interface for Inputting Keywords

金子 恭史[▼] 中村 聡史[◆]
大島 裕明[◆] 田中 克己[◆]

Yasufumi KANEKO Satoshi NAKAMURA
Hiroaki OHSHIMA Katsumi TANAKA

本稿では、検索クエリ内の各キーワードに「緩和度」という要素を加えることで、ユーザの意図を反映した検索結果を提供する手法を提案する。本手法ではまず、ユーザがキーワードを入力する際、そのキーワードの入力速度を調整したり、「？」や「とか」などの補助キーワードを付加したりすることで、ユーザのそのキーワードに対する緩和度を決定する。そして、Web 検索を用いてキーワード間の関係を取得し、その関係と緩和度を用いてユーザがどのような検索結果を望んでいるかという意図を推定する。さらに、その意図を基に、もとのクエリで得られないページを補完するクエリ群を生成し、それらのクエリによって得られた検索結果を意図に応じてマージし、並び替えることにより、ユーザの検索意図を反映した検索結果を返すことを可能にする。

This paper proposes a method of detecting a users' search intention by *relaxation value*. First, the users determine the *relaxation value* by adjusting the speed of inputting keywords or adding *support word* like “?” or “etc”. Second, the system acquires the relationship between keywords using Web search and gets relational keywords of each keyword by the *relaxation value* and the obtained relationships. Third, the system creates several queries which are different from the original query and merges the Web search results obtained by each query. Finally, the system orders and replaces the acquired search result, and show the search result to the users.

1. はじめに

近年、情報検索をする際にキーワード入力型のWeb 検索エンジンを利用することが一般化している。ユーザは求める情報が記述されたページが含んでいそうなキーワードをもとにクエリを作り入力することでその情報にたどり着こうと

する。しかし、そうした適切なクエリを生成することは容易なことではなく、クエリによっては検索結果数が膨大になり過ぎたり逆に絞り込み過ぎたりして対象とする情報を含むページを得られないことが多い。この大きな要因として、既存の検索エンジンにはクエリ中の各キーワードに対する意図を伝えることができないという点が挙げられる。

例えば、「京都らしい豆腐や湯葉を使った和食料理を食べに行きたい」と考え、「京都 豆腐 和食」というクエリで検索を行ったとする。ユーザは京都の和食の店を探しているため「京都」と「和食」を外すことはできないが、「豆腐」に関しては「湯葉」や「豆乳」などでもよいと考えている。しかし、検索エンジンにはそうしたユーザの意図がわからないため、ただ単に「京都」「豆腐」「和食」を3つとも含むページを検索結果として返すだけである。もし、ユーザが「湯葉」や「豆乳」などを含むページも探したいと思った場合には、新たにクエリを作り検索する必要がある。

なお、このクエリの場合、「京都の豆腐料理にこだわりたい」や「京都かその周辺で豆腐を使った和食のお店に行きたい」といった目的で使われることもあると思われる。以上のようにユーザの意図は多様であり、検索エンジンにとって与えられたクエリのみからユーザの意図を反映した検索結果を返すことは困難である。ユーザが自身の意図に沿った検索結果を得るためには、クエリの他にユーザの意図を検索エンジンに伝える仕組みが必要となる。

そこで本研究では、クエリ中の各キーワードに「緩和度」という要素を付加し、それによりユーザの意図をシステムに伝達することを可能とするインタフェースを実現する。ここで、「緩和度」とはそのキーワードをどの程度他の語に置き換えて検索したいかを表す指標であり、これをシステムに伝達する手段としてそのキーワードの入力速度を利用する。また、緩和度による意図伝達を支援するものとして「補助ワード」を用いる。ユーザは補助ワードを各キーワードに付け加えることで伝えた緩和度の補正や、より明示的な意図の伝達ができる。さらに、与えられた緩和度や補助ワードとキーワード間の関係をもとにユーザの検索意図に沿った関連語を推定し、それらを基に複数のクエリを生成し、それらのクエリから得られた結果を組み合わせてユーザの意図に沿った検索結果を取得する。そして、検索結果は緩和度に基づいて並び替えてユーザに提供する。本提案システムにより曖昧で自信のないキーワードもクエリに利用できるようになり、クエリ生成の難易度を下げることができる。また、同一のクエリからでもその時のユーザの意図に応じて異なる検索結果を取得することが可能となる。

2. 関連研究

クエリとして与えられた複数のキーワード間に成り立つ関係を利用した研究は盛んに行われている。野田らはWeb検索を利用して接続助詞「の」で結ばれる関係にあるかを調べることで、クエリ中のキーワードで主題を表すものと話題を表すものを判別する手法を提案している[1]。また、大島らはあるキーワードに対して、その語と接続助詞「や」や「やら」などで結ばれる関係にある語を抽出する手法を提案している[2]。本研究ではこれらの手法を基にキーワード間の関係を抽出する。田らはクエリ中の2キーワード間で成り立つ関係に応じて検索結果をランキングする手法を提案している[3]が、これはもとのクエリによる検索結果に対する最適化であり、本手法はもとのクエリ以外の検索結果も含めてランキングし最適化を行う。

[▼] 学生会員 京都大学大学院情報学研究科 修士課程
kaneko@dl.kuis.kyoto-u.ac.jp

[◆] 正会員 京都大学大学院情報学研究科社会情報学専攻
{nakamura, ohshima, tanaka}@dl.kuis.kyoto-u.ac.jp

ユーザが自ら検索意図を指定できるようなものは多数存在している。Yahoo! Mindset¹では、各ページがあらかじめ Researching と Shopping の二軸で重み付けされており、ユーザが自身の検索意図が Researching よりであるか Shopping よりであるかをスライドバーを用いて指定することで、システムは指定された重み付けに応じた検索結果を返す。また、吉田らは検索時にそのクエリと関わりのある語を自動抽出してグラフ上に複数配置し、ユーザが興味をもった語があれば、それを操作することでその語の配置に応じた検索結果を得ることのできるシステムを提案している[4]。山本らのシステム[5]は、ユーザが任意の語に対して強調または削除の編集操作を行うことで、それに応じたリランキングを行った検索結果を返す。これらの研究はすべて得られた検索結果をもとに主にマウス操作でユーザに意図を伝えてもらい、その意図を反映した検索結果を返すものである。一方で本研究は検索開始時にキーボード操作のみでユーザの意図を伝えるものである。

与えられたクエリを緩和することで検索の精度を上げる研究として、桑原らの研究[6]が挙げられる。これはもとのクエリ中のキーワードを基に複数のクエリを生成し、これを基に Web 検索や画像検索を行い結果を取得する。これに対して本研究では、クエリ中のキーワード以外の語を利用してクエリを生成し、Web 検索を利用して結果を取得している。

3. 緩和度を伝えるインタフェース

3.1 緩和度

本研究でいう緩和度とは各キーワードに与えられる 0 から 1 までの連続値で、値が大きくなるほど緩めて検索するものとする。また、本研究ではこの緩めるということを、対象となるキーワードを関連語で置き換えることにより実現する。また、検索結果のランキング時にそのキーワードの重みを小さくする。

3.2 入力速度による緩和度の伝達

各キーワードに対する緩和度を設定する手段を考える際に、以下の点を考慮する必要がある。

- キーボード操作とマウス操作の切り替えを抑える。
- 特殊な入力ボックスは用いない。
- 特殊な形式の入力を行わせることは避ける。

例えばキーワードに緩和度を設定する手法としてキーワードごとにスライドバーを用意し、マウスで随時値を指定するといった操作が考えられる。しかし、こうした持ちかえは入力の際にユーザの手間を増やすことになるので好ましくない。また、緩和度の高さにより入力ボックスを分けるということも考えられるが、この場合、入力方法が直感的に分かりづらくなってしまいうため、単一の入力ボックスの方が好ましい。一方、緩和度の設定の際に、キーワードの後ろに緩和度を「京都(0.0) 豆腐(0.8) 和食(0.2)」のように入力させる、といった特殊な形式の入力を行うことも考えられる。しかし、こうした入力にはユーザに煩雑な入力を強いることになり、緩和度の明示的な値もキーワード入力のためにユーザに考えさせることになるため、結果としてユーザの負担を増加させてしまう。また、White らの調査[7]によると、「site:」や「-」といったような特殊な検索オプションを使うユーザは検索エンジンを利用するユーザのわずか 8.72% にすぎない。これらのことから、キーワードごとに括弧書きで数値を与え

表 1 補助ワード一覧

補助ワード	効果
!	緩和度を小さくする
?	緩和度を大きくする
とか	キーワードの関連語をもとの語より優先する
かな	もとのキーワードをその関連語よりも優先する

るといった形式の入力は現実的ではない。

これらを踏まえ、我々は緩和度の伝達手段としてキーワード入力時の入力速度の利用を提案する。ここではユーザには緩和度を高くしたいキーワードを他の自信のあるキーワードよりもゆっくり入力してもらう。システムはユーザの入力に従い各キーワードに対して緩和度を設定していく。これを利用することで、通常の入力ボックスにキーボード操作のみで緩和度を伝えることができ、先ほど挙げた 3 つの要件を満たすことができる。

入力速度から緩和度への変換はクエリ全体の平均の入力速度に対するキーワードの平均の入力速度の割合をもとに行う。そのため入力速度の個人差による影響はほとんど受けない。また、通常よりもゆっくり入力するということは操作の難易度自体も高くはないと考えられるので、ユーザに入力速度を用いて緩和度をシステムに伝えてもらうことは十分に可能であると思われる。なお、緩和度については次節で述べる補助ワードによっておおまかな補正を行うことが可能である。

3.3 補助ワードの利用

補助ワードとは、伝達した緩和度を補正したり、より明示的な意図を伝えたりしたいときにキーワードの後ろに付加する語または記号を指す。種類と効果については表 1 に示す通りで、「!」「?」は緩和度の補正に、「とか」「かな」はより明確な意図伝達に用いられる。補助ワードの入力は特殊なものとなるが、緩和度と違いすべてのキーワードに与えられる必要はないため大きな負担を与えるものではない。

補助ワードの利用例を挙げる。「京都 豆腐 和食」というクエリに対して、例えば「京都らしい和食料理を食べに行きたいが、豆腐しか思い浮かばない。ほかに有名なものがあればそれについてもっと知りたい」といった意図のとき、豆腐よりも湯葉や豆乳を優先させた方が望ましい。そういったときに、関連語を優先させる補助ワード「とか」を利用して「京都 豆腐とか 和食」というクエリをシステムに渡すことで、より望ましい検索結果を得ることができると考えられる。

4. キーワード間の関係と関連語の取得

4.1 キーワード間の関係

ここでは単純な言語的手法で取得できる 2 種類の 2 キーワード間の関係を利用する。3 つ以上のキーワード間の関係は 2 キーワード間の関係の組み合わせとみなして処理を行う。

従属関係:「京都 紅葉」、「iPod 価格」のように、キーワード A , B が「 A の B 」という形で表せるとき、それらには従属関係があるとする。 A という主題に対して興味の対象を限定するための話題として B が用いられることが多い。そのため、 A の関連語を得る際に B を考慮することで同じ話題をもつ関連語を得ることができ、逆に B の関連語を得る際に A を考慮することで同じ主題の関連語を得ることができる。

並立関係:「Wii PS3」、「晩婚化 少子化」のように、キーワード A , B が「 A や B 」、「 A と B 」という形で表せるとき、それらには並立関係があるとする。 A と B を比較する場合や、

¹ <http://mindset.research.yahoo.com/>

² <http://developer.yahoo.co.jp/search/>

A と B がともに興味の対象である場合にクエリに含まれることが多く、こういった語の組に対して A にも B にも共起しやすい語が存在することが多い。そのような語はユーザの意図に沿ったものである可能性が高い。一方の関連語を得る際にもう一方の語を考慮することで関連語としてそのような語を得やすくなる。

4.2 キーワード間の関係の抽出

2つのキーワード間の関係を取得するために、2つのキーワード T_1 , T_2 と接続助詞 W_c に対して、 T_1 と T_2 を W_c で結んだフレーズがどの程度妥当であるかを表すスコアを以下の $RelScore(T_1, W_c, T_2)$ を用いて求める。

$$RelScore(T_1, W_c, T_2) = \frac{DF("T_1W_cT_2")}{DF("T_1W_c")} * \frac{DF("T_1W_cT_2")}{DF("W_cT_2")}$$

ここで $DF("X")$ はクエリ " X " によるフレーズ検索の結果ページ数、 $T_1W_cT_2$, T_1W_c , W_cT_2 はそれぞれその順序で語をつなぎ合わせたフレーズを表す。例えば、「京都」と「豆腐」が「の」で結ばれるような関係かを測る場合、評価式の右辺第一項は「京都の」を含むページ中で「京都の豆腐」がどの程度存在するかの割合を表し、右辺第二項が「の豆腐」を含むページ中で「京都の豆腐」を含むページがどの程度存在するかの割合を表す。そのためこれらの積で求められるスコアは「京都の豆腐」がどの程度一般的に利用されるフレーズであるかを表す。

2つのキーワード A , B 間の関係を取得する際には W_c として、「の」、「と」、「や」を用いて、 A , B と B , A の2通りの順序に対してそれぞれスコアを求める。こうして得られた6つのスコアのうち、最も値の高かった関係をその2つのキーワードが満たす関係とする。ただし、これには閾値 L_R を設け、もし閾値以上のものがなければ関係は存在しないものとする。これをキーワードの組み合わせの数だけ行い、各キーワードが関係をもつキーワードを取得する。もしあるキーワードと関係をもつキーワードが複数存在する場合は、その中で最もスコアの高い関係のキーワードのみを利用する。

4.3 キーワードの関連語の取得

本研究ではキーワードの関連語としてそのキーワードの同位語を利用する。同位語とは、共通の上位語を持つような語のことである。例えば「Windows」と「Mac」といったような関係の語である。同位語関係のあるキーワードを発見する手法としては、数回の Web 検索により同位語を発見する大島らの手法[2]やクエリログから発見する山口らの手法[8]などがある。本手法では比較的早く同位語を取得できる大島らの手法を用いる。大島らの手法では同位語を取得する際にキーワードとは別に背景語というものを設定できる。背景語とは欲しい関連語を限定するための語で、これを設定するとともにキーワードのうち、背景語と関係のある関連語を優先的に取得できる。例えば、「木村拓哉」というキーワードのみから同位語を取得すると、「山田監督」「浜崎あゆみ」などが得られる。一方で、背景語として「SMAP」という語を入れると、「香取慎吾」「稲垣吾郎」といったように背景語の「SMAP」に関係のあるものを得ることができるようになる。ここではキーワードの緩和度とキーワードが関係を持つ他のキーワードをもとに適切な背景語を取得し、その背景語を用いて緩和度が反映された関連語の取得を行う。

まず、各キーワードの背景語を取得する。ここで、緩和度に閾値を設け、便宜上緩和度の値に応じて小さいものから順に緩和度の度合いを「なし」「小」「中」「大」の4段階で表す。キーワード A が B と関係をもっているとき、 B の緩和

度が「小」または「なし」ならば A の背景語に B を加える。また、取得する関連語の数は取得数の上限を設け緩和度に基づいて設定する。

以上のようにして設定した背景語と関連語取得数をもとに大島らの手法を用いて各キーワードの関連語を取得する。

5. クエリ群の生成と検索結果の提示

5.1 クエリ群生成

各クエリはもとのクエリのキーワードを関連語で置き換えるか取り除くことで生成される。もとのクエリ q に含まれる n 個のキーワードを $\{k_1, k_2, \dots, k_j, \dots, k_n\}$ とし、 k_j の関連語を $RT(k_j)$ とする。また、 k_j を置き換える候補集合を $Cand(k_j)$ とする。 $Cand(k_j)$ は k_j , $RT(k_j)$ を含み、もし k_j の緩和度が大きければ、その語を含まないクエリを得るために「」（空白）を加える。こうして得られた各 $Cand(k_j)$ の要素を1つずつ含むクエリを組み合わせの数だけ生成する。 m 個のクエリが得られたとき、それらのクエリの集合を $Q = \{q_1, q_2, \dots, q_i, \dots, q_m\}$ とする。

5.2 検索結果集合の取得

前節で得られた各 q_i の検索結果を組み合わせる最終的な検索結果を得る。 P_{total} を総検索結果取得数、 $Count(Cand(k_j))$ を $Cand(k_j)$ の要素数とするとき、それぞれのクエリの検索結果の取得数 $PageCount(q_i)$ を以下のように定める。

$$PageCount(q_i) = P_{total} * \prod_{j=1}^n \frac{1}{Count(Cand(k_j))}$$

検索結果を組み合わせる際には、まず各 q_i の検索結果を組み合わせる。各クエリ q_i に対して、 q_i の検索結果のうち、ほかのクエリでまだ得られていないページを上位から順に $PageCount(q_i)$ 件取得していく。これにより後に検索結果を取得するクエリほどすでに得た検索結果と重複する可能性が高くなるため、より下位のページまで含みやすくなる。そのため、もとのクエリに近いクエリほど後で検索結果を取得するようにする。こうしてすべてのクエリの検索結果を組み合わせることで最終的な検索結果集合を得る。

5.3 検索結果の並び替え

まず、緩和度と補助ワードをもとに各キーワードに重みを設定する。各キーワード k_j の重みの初期値 $w_{base}(k_j)$ を、その k_j の緩和度 $RV(k_j)$ を用いて以下のように定める。

$$w_{base}(k_j) = 1 - RV(k_j)$$

補助ワード「とか」「かな」が付加されたキーワードは、それに応じてさらに重みが調整される。

次に各キーワードの重みをもとに各ページにスコアを付けていく。各ページ p には、そのページを得たクエリ q_i が1つ以上存在する。そのそれぞれのクエリに対して以下のスコア $SubScore(p, q_i)$ を求める。

$$SubScore(p, q_i) = \frac{S_{query}(p) + S_{summary}(p, q_i)}{Rank(p, q_i)}$$

$S_{query}(p)$, $S_{summary}(p, q_i)$, $Rank(p, q_i)$ はそれぞれ、ページ p をクエリ q_i で得た時のクエリに含まれるキーワードの重みの総和、サマリに含まれるキーワードの重みの総和、ページの順位を表す。ここで、あるページ p に対して最も大きな $SubScore$ を得られた q_i を q'_{max} とする。これをもとに、ページ p のスコア $Score(p)$ を以下のようにして求める。



図1 実行例

表2 指定した入力速度

	京都	豆腐	和食
1	ふつう	ゆっくり	ややゆっくり
2	ゆっくり	ふつう	ややゆっくり
3	ふつう	ふつう	ゆっくり
4	ややゆっくり	ゆっくり	ふつう
5	ゆっくり	ややゆっくり	ふつう
6	ふつう	ふつう	ふつう

表3 意図的な入力速度と緩和度

	ゆっくり	ややゆっくり	ふつう
緩和度	0.64417	0.50520	0.06015

$$Score(p) = \frac{S_{title}}{Rank(p, q_{max})} + SubScore(p, q_{max})$$

$S_{title}(p)$ はページ p のタイトルに含まれるキーワードの重みの総和を表す。なお、 $SubScore$ や $Score$ を計算する時に $Rank$ で割っているのは各クエリの検索結果の上位に複数のクエリによる結果を含ませるためである。この補正による影響は上位のページほど大きくなり、最終的なランキングの上位に単一のクエリの検索結果が偏って出現しにくくなっている。

この $Score(p)$ の値の高いページから順に上位に配置することでランキングを行い、その結果をユーザに返す。

6. 評価

6.1 実装

システムのプロトタイプを C#を用いて実装した。検索結果の取得には Yahoo! Japan 検索 API²を利用した。実行例を図1に示す。

入力ボックスに速度を調節してクエリを入力することでランキングされた検索結果を提示する。キーワード入力時には入力ボックスの下に各キーワードがゆっくり入力されたものほど小さく表示される。これによってユーザは意図が正確に伝わっているか確認しながら入力を行うことができ、もしうまく伝えられていなかった場合は補助ワード「!」や「?」で補正することで訂正を行うことが可能である。各ページにはタイトル、サマリ、そのページのスコア、そのページを得たクエリの情報が付加されている。また、右上のコンボボックスから検索に用いられたクエリ群を確認することができ、その中からクエリを選択すればそのクエリの検索結果を上位に置き換えることも可能となっている。

6.2 入力速度による緩和度の伝達の検証

表4 各キーワードに与えられた緩和度の度合い

クエリ	京都	豆腐	和食
クエリ1	なし	大	小
クエリ2	なし	なし	大
クエリ3	大	なし	小

表5 緩和度を利用した場合の結果の差

順位	クエリ1	クエリ2	クエリ3
1位	京都(41)	京都(29)	豆腐(28)
2位	和食(22)	豆腐(29)	料理(16)
3位	中華(19)	中華(7)	和食(12)
4位	豆腐(9)	和菓子(5)	京都(11)
5位	湯葉(8)	イタリアン(5)	奈良(9)
6位	京野菜(8)	中華(4)	大阪(8)

表6 上位20件中新たなクエリから得られたページ

クエリ	クエリ1	クエリ2	クエリ3
得られたページ数	16	13	15
適合ページ数	7	7	14
ページの適合率	0.44	0.54	0.93

入力速度による意図伝達の可能性を検証するため、普段タイピングをよく行う6名の被験者に入力速度をコントロールしながら検索クエリを入力してもらった実験を行った。被験者には「京都 豆腐 和食」というクエリに対して意図的に表2で指定された速度で各キーワードを入力してもらった。得られた緩和度の意図ごとの平均を表3に示す。

緩和度は入力速度と反比例するため、平均よりゆっくり入力するほどその変化が小さくなり、入力速度の構成による緩和度の伝達は難しくなる。しかし表3に表れているように、3語程度なら十分に入力速度を調整し緩和度を伝えることができおり、ユーザが意図的に入力速度を調整することでシステムに大まかな緩和度を伝えることができる可能性は十分にあると言える。

6.3 検索結果の多様性の検証

ここでは本手法によって同じクエリから得られる異なる結果の比較を行う。クエリは「京都 豆腐 和食」を用い、関係を求める際の閾値 L_R は0.1、総取得件数 P_{total} は200件としている。

1章で用いた例をもとにした表4のような3通りの緩和の結果を比較する。各クエリによって得られた結果のそれぞれ上位20件に対して、そのページを得たクエリ中に含まれるキーワードのうち頻出するもの上位6語を取得した。その結果を表5に示す。表5の括弧内の数値は上位20件中に現れたキーワードの個数である。表5からわかるように上位に特定の話題が偏ることなく、ほぼ5.2節で設定した割合で各キーワードの結果が含まれており、期待通りの成果を得られていると言える。

次に、検索意図ごとに上位20件中でもとの「京都 豆腐 和食」というクエリ以外のクエリによって得られたページの適合率を調べた。その結果は表6の通りで、もとのクエリでは得られなかったページを検索結果に含むことに成功していることがわかる。一方適合率に関してはもとのクエリによる検索結果の上位20件の適合率を調べたところ0.9だったのに対し、本手法では0.44から0.93と不安定なものだった。これについてはクエリによる影響も大きいですが、アルゴリズム等を見直して改善する必要があると思われる。しかし本手法

² <http://developer.yahoo.co.jp/search/>

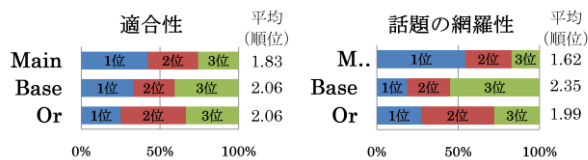


図2 各観点の結果

が用いられるのはキーワードに自信がなく他の候補が思いつかないような場合と考えられる。従って多少適合率を下げてもこのように自動的に話題を広げた検索結果を提供することが求められると思われるので、全体として望ましい結果だと言える。

6.4 ユーザによる主観評価

6.4.1 評価方法

この評価では、本手法によって得られる結果 R_{main} に対する比較対象として、もとのクエリによって得られる結果 R_{base} 、本手法によって得られる各キーワードの関連語をもとのキーワードに Yahoo! 検索エンジンの検索オプション「OR」を用いてつなげることで作られるクエリによって得られる結果 R_{or} を用いる。例えば、もとのクエリが「京都 豆腐 和食」で、豆腐の関連語として「湯葉」、「豆乳」を取得している時、 R_{base} はクエリ「京都 豆腐 和食」によって得られる結果を、 R_{or} はクエリ「京都 豆腐 OR 湯葉 OR 豆乳 和食」によって得られる結果を表す。また、

準備した 20 通りのうち例として 5 つの検索意図とそれに対するクエリを表 7 に示した。表 7 中のクエリ内の各キーワードの後ろの括弧内の文字は、 R_{main} を得る際にそのキーワードに与えられた緩和度の割合を表している。なお、緩和度を低く設定したキーワードの緩和度の表記は省略している。被験者にはこの 20 通りの検索意図を順に提示し、以下に示す 2 つの観点それぞれに対して、その検索意図に基づく 3 つの結果 R_{main} 、 R_{base} 、 R_{or} の各上位 10 件に順位をつけてもらう。

1. 意図に沿ったページを含む割合が高いと思う結果は？
2. 意図に沿った話題をより網羅していると思う結果は？

この評価には普段検索エンジンを頻繁に利用している 20 代のユーザ 7 名に被験者として参加してもらい、それぞれの検索意図に対して、その検索意図、クエリ、観点、3 つの比較対象となる結果を提示し、観点に沿うものから順に対応する結果のボタンを押してもらうことで順位付けを行ってもらった。なお、3 つの結果はランダムに並び替えられてユーザに提示される。

6.4.2 評価結果

それぞれの観点に関して、被験者が各結果に与えた順位の全体の平均を図 2 に示す。

まずは図 2 の観点 1 の結果から本手法の有用性を検証する。 R_{main} を R_{base} と比較したとき、より多くの話題を扱うだけ R_{main} の結果に不必要なページが含まれると思われたが、実際には本手法の方が有効であったことが読み取れる。また、 R_{main} を R_{or} と比較した場合にも同様の結果が得られている。これは単一のクエリの結果を下位まで表示するよりも、関係のありそうなクエリの上位のみを取り出して表示させた方がユーザの意図に沿った結果を得られるということを示しているのではないと思われる。

次に図 2 の観点 2 の結果をもとに本手法の有用性を検証す

る。 R_{main} を R_{base} と比較すると、明確な差が表れている。これにより本手法で取得した関連語によって広げられた話題が、実際にユーザの意図に沿った話題であったことが示された。また、検索結果の結合とスコアリングによる並び替えの有用性を検証するために R_{main} を R_{or} と比較すると、本手法の結果の方がより多くの話題を網羅しているという結果が得られた。これは、本手法では各キーワードに対する検索結果の取得割合が調整されており、それぞれの結果の上位ページが順に配置されるようになっているため、上位 10 件の結果を見ても多くの話題が均等に現れやすくなっているのだと思われる。

最後に意図ごとの評価結果について述べる。表 7 の意図 1 ではよい結果が得られなかった。これは、本手法では「豆腐」の関連語として「湯葉」や「豆乳」を用いて話題を広げようとしているが、「京都」というキーワードが含まれている時これらの関連語は非常に共起しやすいため、「豆腐」というキーワードのみで検索しても「湯葉」や「豆乳」といった語を含むページが多く取得され、本手法の有用性が薄れてしまっているものと思われる。このことから、もとのクエリのみによる検索結果にも頻出する共起語を関連語に含めるかについては考慮する必要がある。一方で、意図 5 のように人名のキーワードを緩和している意図については非常に良い結果を得ることができた。他にも意図 3、4 のように、固有名詞のキーワードを緩和して良い評価結果が得られているものは多く、固有名詞のキーワードを緩和するような意図に対して本手法は有効であると言える。また、「最新」「新型」「方法」「評価」「性能」などのような一般的な語に高い緩和度を付加することにより、ユーザは本来のキーワードのみで検索した場合よりも多くの情報を得ることができていた。

6.5 システムの考察

このシステムの長所は、例え自信のないキーワードであってもユーザはそれをクエリに利用できるという点である。さらに、自信のないキーワードが思い浮かんでいるということは、それに代わるような適切なキーワードが思いつかない状態であると考えられる。そのため自動的にキーワードを置き換え検索結果に反映してくれるシステムは、自信のないキーワードを入力しようとするユーザにとって望ましいものであると言える。

我々の手法では、複数のクエリの検索結果を結合して得られた検索結果の場合、なるべく同じクエリで得られたページが連続して現れないようにランキングしている。そのためユーザは、検索結果の上位を見るだけで多様なクエリのページに触れることができる。

一方でこのシステムの問題点として、処理時間が挙げられる。本稿のアルゴリズムだと、キーワードの数が増えるにつれてキーワードの組み合わせの数やクエリ数が膨れ上がってしまい、アクセス数が非常に多くなってしまふ。そのため、不必要なアクセスを減らすことが必要とされる。その他の問題点として、本手法に有効なクエリに限られているという点が挙げられる。今回利用した関連語は同位語のみであるため、緩めたいキーワードがあったとき、その一例が思い浮かばないと利用できない。より幅広いクエリに対応させるには、上位語や下位語なども関連語に含める必要があるが、前述のように実行時間の問題があるので、そのことも考慮して実現可能な手法を考えていかなければならない。

表7 評価に用いた検索結果の例

	検索意図	クエリ (括弧内は緩和度の大きさ)
1	京都に例えば豆腐を使った和食料理を食べに行きたい	京都 豆腐(大) 和食
2	新しいダイエット方法ってないのだろうか	最新(小) ダイエット 方法(大)
3	カシオのデジカメの性能を他のメーカーのものと比べたい	カシオ(大) デジカメ
4	インフルエンザに効く薬ってタミフル以外にはないのだろうか	インフルエンザ タミフル(大)
5	稲垣吾郎と他の SMAP メンバーと一緒に出演しているテレビ番組などを知りたい	稲垣吾郎 木村拓哉(大) テレビ(中)

また、ユーザが入力速度に差をつけることが可能であることは実験によって示したものの、まだ実験数が十分とは言えない。また、ユーザの普段の入力速度はばらつきがあるものであり、さらに、同じ「ゆっくり」入力する場合でも人によってどの程度入力速度を調整するかは異なる。本システムでは緩和度に閾値を定めて大きさの度合いを4段階に分け、その度合いに応じて処理を行っているが、この閾値は単純に0から1までの連続値を等間隔に分割して設定したもので、ユーザの入力の傾向を考慮したものにはなっていない。この緩和度の度合いの閾値については考慮していく必要がある。

7. まとめ

本稿では、入力速度によってユーザの意図をシステムに伝えるインタフェースを実現し、それによる意図伝達が可能であることを示した。さらに、その伝えられた緩和度に応じてシステムがユーザの検索意図を反映した検索結果を返す手法を提案し、その検索結果の有用性も示した。また、補助ワードによる入力インタフェースの補強にも取り組んだ。

今後の課題としては、まずキーワード間で成り立つ関係からより多くの意図推定を行うことが挙げられる。本稿では従属関係と並立関係で処理を分けなかったが、厳密には関係ごとにさらに処理を分けるべきである。また、本稿では関連語として同位語のみを利用したが、検索意図によっては同位語以外の語が求められる場合も考えられるため、他の関連語の取得方法を考えていく必要がある。

今回のシステムでは、入力速度によってキーワードの緩和度を指定していた。しかし本システムは1文字ごとに検索意図を付加させることも可能である。それによって同じキーワードのどの部分に不安があるかをより明確に伝えてもらい、それに応じた結果を返すことも考えられる。また、補助ワードについても、より直感的なものにしていく予定である。

検索結果の閲覧時のインタフェースについては、各ページにはそのページを得る際に用いられたクエリとその時のサマリ、順位が付加されており、これらをもとにフィルタリングやランキングの改良を行うことが考えられる。

【謝辞】

本研究の一部は、文部科学省科学研究費補助金特定領域研究「情報爆発時代に向けた新しい IT 基盤技術の研究」における、計画研究「情報爆発時代に対応するコンテンツ融合と操作環境融合に関する研究」(研究代表者: 田中克己, A01-00-02, 課題番号 18049041)、計画研究「情報爆発に対応する新 IT 基盤研究支援プラットフォームの構築」(研究代表者: 安達淳, Y00-01, 課題番号: 18049073)、および、文部科学省グローバル COE 拠点形成プログラム「知識循環社会のための情報学教育研究拠点」(研究代表者: 田中克己, 平成 19~23 年度)によるものです。ここに記して謝意を表すものとします。

【文献】

- [1] 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”, 情報処理学会研究報告. データベース・システム研究会報告, **2006**, 78, pp. 305-311 (2006).
- [2] 大島裕明, 小山聡, 田中克己: “Web 検索エンジンのインデックスを用いた同位語とそのコンテキストの発見”, 情報処理学会論文誌. データベース, **47**, 19, pp. 98-112 (2006).
- [3] 田馳, 手塚太郎, 小山聡, 田島敬史, 田中克己: “質問キーワードの意味的関連と近接性に着目したウェブ検索の精度改善”, 電子情報通信学会第17回データ工学ワークショップ(DEWS2006) (2006).
- [4] 吉田大我, 小山聡, 中村聡史, 田中克己: “Web 検索結果におけるキーワード出現相関の可視化と対話的な質問変換”, 電子情報通信学会第18回データ工学ワークショップ(DEWS2007) (2007).
- [5] T. Yamamoto, S. Nakamura and K. Tanaka: “Rerank-By-Example: Efficient Browsing of Web Search Results”, Proc. of DEXA2008, pp. 801-810 (2007).
- [6] 桑原昭裕, 角谷和俊, 田中克己: “質問緩和法によるクロスメディア・メタサーチ”, DBSJ Letters, **3**, 1.
- [7] R. W. White and D. Morris: “Investigating the querying and browsing behavior of advanced search engine users”, SIGIR2007, pp. 255-262 (2007).
- [8] 山口雅史, 大島裕明, 小山聡, 田中克己: “サーチエンジンのクエリログを利用した同位語の発見”, DBSJ Letters, **5**, 2.

金子 恭史 Yasufumi KANEKO

京都大学大学院情報学研究科修士課程在学中。主にインタラクションの研究に従事。日本データベース学会学生会員。

中村 聡史 Satoshi NAKAMURA

京都大学大学院情報学研究科社会情報学専攻特定講師。2004年大阪大学大学院情報学研究科博士後期課程修了。博士(工学)。主にインタラクション、ウェブ検索の研究に従事。情報処理学会、日本データベース学会会員。

大島 裕明 Hiroaki OHSHIMA

京都大学大学院情報学研究科社会情報学専攻特定助教。2007年同博士後期課程修了。博士(情報学)。主にウェブ、情報検索の研究に従事。情報処理学会、電子情報通信学会、日本データベース学会、ACM 各会員。

田中 克己 Katsumi TANAKA

京都大学大学院情報学研究科社会情報学専攻教授。1976年京都大学大学院修士課程修了。博士(工学)。主にデータベース、マルチメディアコンテンツ処理の研究に従事。IEEE Computer Society, ACM, 人工知能学会、日本ソフトウェア科学会、情報処理学会、日本データベース学会等各会員。