

対話的に調整可能な文書ランキング——WWW 検索支援の一手法

仲川 ころろ[†], 木下 敦史[†],
高田 喜朗[†] 関 浩之[†]

本研究では, WWW 検索システムが提示するランキング(抽出した文書をスコアの降順に並べたリスト)に対して, ユーザが自分の検索目的に沿うように調整システムはその操作を基に文書のスコア計算規則を修正する, という検索支援手法を提案する. 従来, キーワード入力型検索システムにおける様々なスコア計算方法が研究されているが, 同じキーワードに対して同じ結果を出力する限り, つねに適切なランキングを提供することは難しい. このときユーザは, 欲しい文書を見つけにくいだけでなく, ランキングを調整する手段がないことから, 心理的負担を感じることになる. 本論文ではユーザの調整操作をランキング修正に反映する 3 通りのアルゴリズムを提案する. 試作システムを用いて行った 2 種類の評価実験(被験者による評価実験およびベンチマークテスト)の結果, 調整を行うことによって適合文書が選択的に上位に持ち上げられていることが観測でき, 下位に埋もれた未知の適合文書を発見しやすくするという点で有効な手法であることが確認できた.

User-adjusted Document Ranking — A Supporting Method for WWW Document Retrieval

KOKORO NAKAGAWA,[†] ATSUFUMI KINOSHITA,[†]
YOSHIAKI TAKATA[†] and HIROYUKI SEKI[†]

We propose a method for supporting WWW document retrieval which allows a user to directly adjust a ranking of documents. Many keyword-based search systems are available and these systems provide a user with a ranked list of documents, arranged in a descending order of relevancy to input keywords. Various scoring methods to rank documents have been proposed; however, a highly-ranked documents are not always desirable for a user since users with different purposes may input the same keywords. This difference puts frustration on the user because not only is it difficult to find desirable documents, but also no means to adjust the ranking is provided. We propose three methods for modifying scoring criterion based on the adjustment by a user. A prototype system based on the proposed methods has been implemented. Results on an experiment with six human subjects and on a benchmark test show that the proposed system is useful for finding relevant documents which were ranked low before the adjustment.

1. はじめに

WWW (World Wide Web) の普及と流通する文書の増加にともない, WWW 文書を検索するためのサービスが数多く提供されるようになったが, いまだ問題点も多く, 検索作業におけるユーザの負担は大である.

現在提供されている WWW 検索サービスのほとん

どは, Google のようなキーワード検索型が, もしくは Yahoo! のようなキーワード検索とディレクトリの結合型である. これらのキーワード検索サービスは, ユーザにとって簡便な手法として広く受け入れられているが, 以下のような問題点がある.

(1) 適切なキーワードを考える難しさ

目的の情報だけを的確に取り出すことは一般に難しく, 通常, 簡単なキーワードを入力しただけでは多くのハズレ(ユーザにとっての不適合文書)を含む大量の文書が検索され, 精度の悪い結果しか得られないことが多い. 精度の良い(適合率・再現率の高い)文書

[†] 奈良先端科学技術大学院大学情報科学研究科
Graduate School of Information Science, Nara Institute
of Science and Technology
現在, 独立行政法人通信総合研究所
Presently with Communications Research Laboratory
現在, 三洋電機株式会社
Presently with SANYO Electric Co., Ltd.

<http://www.google.com/>

<http://www.yahoo.com/>

集合を結果として得るためには、適切なキーワードや検索式を指定するか、または精度の悪い一時的な結果を追加キーワードなどによってさらに絞り込んでゆく必要があり、ユーザにとって慎重さを要する作業となる。専門的な知識や技術が必要になる場合もあり、検索作業に不慣れなユーザにとって難しい作業である。

(2) 一方的な結果表示によるストレス

多くのキーワード検索サービスは、検索要求のたびに、データベース中の各文書に対してクエリ(ユーザが入力したキーワードや検索式)との適合度を計算し、適合度が閾値よりも大きい文書を適合度の降順に並べてユーザに提示する仕組みになっている。本論文では、文書とクエリとの適合度をスコア、文書をスコアの降順に並べたリストのことをランキングと呼ぶ。システムが抽出する文書すべてが必ずしもユーザにとっての適合文書ではないので、ユーザは、提示されたランキングを上位から下位へと順にチェックして、検索目的に適合した文書を探すことになる。この際、文書リストはすべてをチェックできないほど長い場合がほとんどであるため、ユーザが実際に見るのは上位からある程度までの文書だけになる。したがって、たとえユーザにとって適合度の高い文書を多く含むリストでも、それらがランキングの上位に位置していなければ、ユーザにとっては精度の悪い結果になる。つまり、ユーザにとっての適合文書が上位に集まった適切なランキングを提供することが重要であるが、入力キーワードから正確な検索目的を把握するのは困難であるので、スコアの計算方法をいかに工夫しても、同じキーワードに対して同じランキングを出力する限りつねに適切なランキングを提供することは難しい。

しかし、現在の検索サービスではこのような場合にユーザがランキングを変更・修正する手段は提供されておらず、ユーザはシステムの提供する不適切なランキングを受け入れ続けなければならない。これにより、適合文書を得るまでに余計な時間や手間を要するのみでなく、ランキングの不適切さをシステムに指示することができないという心理的なストレスも引き起こしていると考えられる。

本研究の目的

上記の問題(1)に対し、ユーザが複雑なキーワードや検索式を考えなくても検索の効率や精度を改善できるような手法がすでに数多く研究されており、たとえば、統計的分類手法や先験的知識(シソーラスやオントロジーなど)を用いて得られた分類結果を提示する手法、自然言語解析やシソーラスを用いたキーワード記述の補助などがある。一方問題(2)については、

	検索目的との適合度
1. score:99 http://www.aaa.com/	✓ yes
2. score:98 http://www.bbb.com/	no
3. score:96 http://www.ccc.com/	no
4. score:90 http://www.ddd.com/	✓ yes
5. score:89 http://www.eee.com/	no
⋮	
11. score:68 http://www.kkk.com/	✓ yes

図1 ユーザによるランキングの調整

Fig.1 Adjustment by the user.

ユーザの検索目的や特徴を自動的に推測し、クエリを適応させていくような研究がある(適合フィードバック)。しかし、著者らの知る限り、ユーザがランキングやスコア計算規則そのものを直接制御・修正できるような手法はない。

本研究では、提示されたランキングに対し、ユーザが任意文書の順位上げ/下げ操作によって自分の適合度に沿うように調整できる、という手法を提案する⁶⁾。これにより上記問題(2)「ユーザが不適切なランキングを我慢し続けなければならない」ことの解決を図る。提案手法では、ユーザはまず従来どおりのランキング出力を得、不当にランクが低いと感じる文書を上位に押し上げたり、逆に自分にとっては不適合文書なのにランクが高い文書を下位に押し下げたりすることで、システムに対して「現在のランキングはこのように不適切である」という指摘を行うことができる(図1)。システムは、ユーザの行った操作に応じてユーザの検索目的を推測し、それに沿った新しいランキングを作成し提示する。この際、ユーザが指示した文書についてのみ順位を変更するのではなく、ユーザの指示した文書を適合/不適合文書のサンプルと考えて現在のスコア計算規則を修正し、ランキング中の全文書についてスコアを再計算する。これにより、ユーザにとっての適合文書が上位に集まったより適切なランキングを作成し提供することを目指している。

1.1 関連研究

ユーザの検索目的を推測し、それに適応することによって文書検索を支援しようという研究は1970年代まで遡ることができ、また近年のWWWや検索システムの普及にともなってこのような研究が多くなされるようになった。ユーザにとっての使いやすさの向上を図るという観点から、従来のWWW検索支援手法と提案手法を比較する。

(A) 文書集合の改良

初期検索の結果を絞り込むのに有用な 2 次キーワードを提供するシステムとして, RCAAU⁵⁾, Anickらのシステム¹⁾, HiB²⁾ などがある. また, 初期文書集合を分類することでより精度の高い部分集合を得ることを目的としたシステムとして, Scatter/Gather¹⁰⁾, HIBROWSE¹¹⁾, CATE^{7),15)} など数多くの研究がなされている. これらの研究は, 精度の悪い初期検索の結果を絞り込んでより精度の良い文書集合を得ることを目的としており, 文書集合の提示段階においてユーザの負担を軽減しようとする本研究とは異なるアプローチである. しかし, 不適切な文書がほとんどなくなるまでユーザが検索式を精練することは難しいため, 絞り込みによる検索支援手法においても適切なランキングを提示するための技術は必要である. この点から提案手法と上記検索支援手法との統合が期待される.

(B) 適合フィードバック

適合フィードバック¹³⁾では, 現在の検索式で得た文書集合中の各文書に対し, ユーザが適合・不適合の判定を行い, システムはそれを基に検索式を洗練する.

本研究の提案手法では, 調整作業によってスコアが再計算されるだけで文書集合は変更されない. 一方, 適合フィードバックでは, 検索式が変更されるたびに文書集合とスコアの両方が変更されるが, 文書集合はそのままスコアだけを変更するように実装することも可能である. この場合, ユーザの検索目的に適應した新しいスコア付けを目指すという点で, 提案手法と適合フィードバックは似たアプローチになる.

AIQEC³⁾ではユーザが判定作業をクラスタ単位で行えるようにすることで負担の軽減を目指しているが, 各クラスタの内容を簡潔に表現するラベルを付けることは難しいため, 判定作業の難しさが増すことも考えられる. 提案手法では, ユーザは文書の順位の上げ/下げという単純な操作を行うだけでよい. さらに, 現在の WWW 検索サービスにおいて最終的な出力はつねにランキング形式であるため, 検索サービスのユーザは文書空間に対してベクトル空間モデルのような空間上の分布ではなくスコア順に並んだリストに近いメンタルモデルを持っていると考えられる. ユーザにとって精度の悪いリストとは, “正しい順序に並んでいないだけでどこか(下位)には適合文書がまぎれている” ようなものであり, 提案手法の提供する「このような文書は上位に移動せよ」という指定方法は上記のメンタルモデルに沿った無理のない表現形式と考えられる.

(C) スコア計算方法の改良

基本的に文書のスコアとは, 入力されたキーワード

(検索式)とその文書との類似度であるが, 検索サービスの多くは, 単純な類似度だけでなく $tf * idf$ ¹⁷⁾ や文書中の標題・見出しなどへの重み付け, PageRank⁹⁾ 技術などを用いてスコアの改良を図っている. これらのスコア改良技術は, 本研究の提案手法や上記 (A), (B) で述べた検索支援手法との組み合わせが可能である.

2. 提案手法の概要とシステム設計の方針

2.1 システムの動作概要

システムの動作手順は以下ようになる:

- (1) ユーザがキーワードを入力する.
- (2) システムは, データベース中の各文書ごとにキーワードに対する適合度を計算し, その降順に並べた文書リスト(初期ランキング)を出力する.
- (3) ユーザは, 不当に順位が低いと感じる文書を上位に上げるか, または自分にとって不適合と感じる文書を下位に下げる操作を行う(図 1).
- (4) システムは, ユーザの調整操作(文書順位の上げ/下げ)からユーザの調整意図を推測し, スコア計算規則を変更する(3章).
- (5) システムは, 手順(4)で推測した新しい計算規則を用いて, 手順(2)で抽出した各文書のスコアを再計算し, 新しいランキングをユーザに提示する.

1 回の調整作業ごとにランキング中の全文書のスコアが再計算されるため, ユーザが直接調整しなかった文書も順位が変化することになる. ユーザにとっての適合度を手順(4)でうまく推測できれば, 新しいランキングでは適合文書が(調整前よりも)上位に集まると期待できる.

以下, 本手法の各要素について説明する.

2.2 ランキングの作成と調整

提案手法では, 文書の表現方法やスコアを定義するためにベクトル空間モデル¹⁴⁾を用いている. 有限集合 S に対し, S の要素数を $|S|$ と書く. ベクトル v_1 と v_2 の内積を (v_1, v_2) , ベクトル v の長さを $|v| = \sqrt{(v, v)}$ とそれぞれ表記する.

ユーザが手順(1)で入力したキーワードの集合を $KW = \{kw_1, kw_2, \dots, kw_p\}$ とすると, 文書 d の KW に対する縮約特徴ベクトル d は以下で定義される $p + m$ 次元ベクトルである.

$$d = \underbrace{(c_{kw_1, d}, \dots, c_{kw_p, d})}_p, \underbrace{(c_{f_1, d}, \dots, c_{f_m, d})}_m \quad (1)$$

ここで, $c_{kw_u, d}$ ($1 \leq u \leq p$) はキーワード kw_u と文書 d の関連度 ($tf * idf$ ¹⁷⁾ など), $c_{f_v, d}$ ($1 \leq v \leq m$)

は単語以外の文書の特徴を示す値(たとえば文書中にあるリンクの数や更新の新しさなど)である。

2つのベクトル v_1 と v_2 間の類似度 $\text{sim}(v_1, v_2)$ を以下のように2種類定義する。

$$\text{sim}(v_1, v_2) = (v_1, v_2) \quad (2)$$

$$\text{sim}(v_1, v_2) = \frac{(v_1, v_2)}{\|v_1\| \|v_2\|} = \cos \theta \quad (3)$$

式(2), (3)は, どちらもベクトル空間モデルにおいてよく利用される定義である。提案手法では3章で述べる推測手法によってこの2種類の類似度を使い分ける。

システムが手順(4)で推測するスコア計算規則とは, 縮約特徴ベクトルの各成分に対する重みを成分に持つベクトル $k = (k_1, k_2, \dots, k_{p+m})$ (クエリーベクトルと呼ぶ)である。 k の x 番目の成分 k_x は, d の x 番目の成分に対する重みを示している。システムが手順(2)で抽出した文書集合を D , ユーザの i 回目の調整操作に応じてシステムが作るクエリーベクトルを $k^{(i)}$ とする。 i 回目の調整後の文書 d のスコア $\text{score}(d, i)$ ($i \geq 0$)を,

$$\begin{aligned} \text{score}(d, 0) &= \text{sim}(d, k^{(0)}) \\ \text{score}(d, i) &= \text{sim}(d, k^{(i)}) + \alpha (\text{sim}(d, k^{(i)}) \\ &\quad - \text{sim}(d, k^{(i-1)})) \end{aligned} \quad (4)$$

とする(α は, 今回の調整による変化をどの程度強調するかを決める非負定数。 α が0の場合, $\text{score}(d, i)$ は $\text{sim}(d, k^{(i)})$ と等しい)。ユーザの i 回目の調整後の D のランキングとは, 以下の不等式を満たす文書リスト $d_1, d_2, \dots, d_{|D|}$ である。

$\text{score}(d_1, i) \geq \text{score}(d_2, i) \geq \dots \geq \text{score}(d_{|D|}, i)$ 。
ただし, 初期ランキングの作成(手順(2))には $k^{(0)} = (\underbrace{1, 1, \dots, 1}_p, \underbrace{0, 0, \dots, 0}_m)$ で表される初期クエリーベクトルを用いる。 $k^{(i)}$ ($i \geq 1$)の作成方法を3章で述べる。

3. クエリーベクトルの推測手法

2.1節手順(3)において, ユーザが文書 d_h を文書 d_l ($1 \leq l < h \leq |D|$)のすぐ上に移動する調整操作を行った場合, ユーザの意図を次のように仮定する。

- d_h は d_1, d_2, \dots, d_{l-1} よりも検索目的に適していない。

- d_h は $d_l, d_{l+1}, \dots, d_{h-1}$ よりも検索目的に適している。

上記の意図を反映する新しいクエリーベクトルを作成するために, 本研究では, 以下に述べる3種類のクエリーベクトル推測手法を提案する(3.1~3.3節)。ただしこれらの推測手法は, 必ずしも文書 d_h をちょうど d_{l-1} と d_l の間に配置するとは限らない。

3.1 成分補正法(手法A)

文書 d_i の縮約特徴ベクトル第 j 成分 ($1 \leq j \leq p+m$)を $d_{i,j}$ と書く。すなわち $d_{i,1} = c_{kw_1, d_i}, \dots, d_{i,p+m} = c_{f_m, d_i}$ (式(1)参照)。また $d_l, d_{l+1}, \dots, d_{h-1}$ の平均ベクトルを d_{lows} とする。すなわち, $d_{lows} = \sum_{i=l}^{h-1} d_i / (h-l)$ 。 d_{lows} の第 j 成分も $d_{lows,j}$ と表す。本手法では, 新しいクエリーベクトルの第 j 成分 $k_j^{(i)}$ を, 現在のクエリーベクトルの第 j 成分 $k_j^{(i-1)}$ と, 2つの縮約特徴ベクトル d_{lows}, d_h の第 j 成分の差 $d_{h,j} - d_{lows,j}$ に基づいて決定する。具体的には, まず, スカラー y を未知数とする次のような1次不等式を考える。

$$\begin{aligned} \text{sim}(md_h, k^{(i-1)} + y(md_h - md_{lows})) &\geq \\ \text{sim}(md_{lows}, k^{(i-1)} + y(md_h - md_{lows})) \end{aligned} \quad (5)$$

ここでは類似度として式(2)を用いる。また, md_h, md_{lows} はそれぞれ d_h, d_{lows} に対して後で述べる補正を行ったベクトルである。式(5)は, 補正後のベクトル md_h, md_{lows} において, md_h と $k^{(i-1)} + y(md_h - md_{lows})$ の類似度が md_{lows} と $k^{(i-1)} + y(md_h - md_{lows})$ の類似度以上になることを意味している。そして, 式(5)を満たす最小の y に対して,

$$k^{(i)} = k^{(i-1)} + y(md_h - md_{lows})$$

を新しいクエリーベクトルとする。

補正について

$d_i = (d_{i,1}, d_{i,2}, \dots, d_{i,p+m})$ に対し, ベクトル md_i を以下のように定義する。

$$md_i = (\eta_1 d_{i,1}, \eta_2 d_{i,2}, \dots, \eta_{p+m} d_{i,p+m})$$

すなわち md_i は, d_i の第 j 成分に下記の補正係数 η_j を掛けたものである。 d_{lows} に対しても同様に md_{lows} を定義する。 η_j ($1 \leq j \leq p+m$)は次のように定義する。

$$\eta_j = \begin{cases} 0 & \text{条件 (a) が真} \\ 0.5 & \text{条件 (b) が真} \\ 1 & \text{otherwise} \end{cases}$$

手順(1)でユーザがキーワード以外の要素についても指定できるように実装すると, 後半の m 成分についても0以外の数値を持ちうる初期クエリーベクトルになる。5章の実験では5.1節で述べるような初期クエリーベクトルを用いた。

以下, 特別な場合として $l=1$ のときは $d_{l-1} = d_h$ と見なす。

$$(a) = \begin{cases} d_{h,j} > d_{lows,j} \wedge d_{l-1,j} < d_{lows,j}, \text{ or} \\ d_{h,j} < d_{lows,j} \wedge d_{l-1,j} > d_{lows,j} \end{cases}$$

$$(b) = \begin{cases} d_{h,j} > d_{lows,j} \wedge d_{l-1,j} < \frac{d_{lows,j} + d_{h,j}}{2}, \text{ or} \\ d_{h,j} < d_{lows,j} \wedge d_{l-1,j} > \frac{d_{lows,j} + d_{h,j}}{2} \end{cases}$$

この補正は、クエリーベクトルを作成する際に、ユーザが変更を希望していないと思われる成分をシステムが誤って大幅に変更してしまうことを避けるために行う。すなわち：

- 条件 (a) が真となるのは、第 j 成分に関して、上位に移動した d_h が d_{lows} より値が大きいにもかかわらず、 d_{l-1} が d_{lows} よりも小さい場合、もしくは d_h が d_{lows} より値が小さいにもかかわらず、 d_{l-1} が d_{lows} よりも大きい場合である。この場合、ユーザの調整意図は第 j 成分とは無関係と考え、補正係数 η_j を 0 とする。
- 条件 (b) が真となるのは、第 j 成分に関して、 d_{l-1} が d_h よりも d_{lows} に近い場合である。この場合、ユーザの調整意図と第 j 成分の関係は、 d_{l-1} が d_{lows} よりも d_h に近い場合 ($\eta_j = 1$ の場合) に比べて薄いと見え、補正係数を $\eta_j = 0.5$ とする。

3.2 E 尺度に基づく推測手法 (手法 B)

3章の冒頭で述べたとおり、ユーザが i 回目の調整で文書 d_h を文書 d_l ($1 \leq l < h \leq |D|$) のすぐ上に移動した場合について考える。本手法では次の2つのベクトル g (good) と b (bad) を用いる。

$$g = (d_{l-1} + d_h) / 2$$

$$b = d_{lows} = \sum_{i=l}^{h-1} d_i / (h-l) \quad (3.1 \text{ 節})$$

直観的に、新しいクエリーベクトル k は、できるだけ g に類似し、かつ、できるだけ b に類似していないことが求められる。そこで本手法では、ベクトル k, g の類似度、および k, b の非類似度を定義し、それらの調和平均を最大にするような k を求める。この手法は、検索結果の精度を評価するための指標の1つである E 尺度¹⁶⁾ から発想を得ている。 E 尺度は、検索結果の適合率 P と再現率 R の重み付き調和平均から求められる。

本手法では類似度として式 (3) を用いる。また、ベクトル間の非類似度 $\text{dsim}(v_1, v_2)$ を以下のように定義する。

$$\text{dsim}(v_1, v_2) = \sqrt{1 - \left(\frac{(v_1, v_2)}{\|v_1\| \|v_2\|} \right)^2} = \sin \theta \quad (6)$$

次に、与えられたベクトル g と b に対して、以下の r を最大にするような k を求め、 $k^{(i)}$ とする。

$$r = \frac{\text{sim}(g, k) \text{dsim}(b, k)}{\text{sim}(g, k) + \text{dsim}(b, k)} \quad (7)$$

式 (7) の r を最大にする長さ 1 のベクトル k は、次式 (8) で与えられる (導出過程は付録参照)。

$$k = \frac{c g - b}{|c g - b|} \quad \text{ただし} \quad c = \frac{1 + \sin \theta}{\cos \theta} \quad (8)$$

3.3 順序情報に基づく推測手法 (手法 C)

3.1, 3.2 節と同様に、ユーザが i 回目の調整で文書 d_h を文書 d_l ($1 \leq l < h \leq |D|$) のすぐ上に移動した場合について考える。3章冒頭で述べた条件を満たすランキングを生成するために、下記の不等式を満たすクエリーベクトル $k = (k_1, \dots, k_{p+m})$ を求める。ただし、本手法では類似度として式 (2) を用いる。

$$\text{sim}(d_h, k) \leq \text{sim}(d_j, k) \quad \text{for } 1 \leq j \leq l-1 \quad (9)$$

$$\text{sim}(d_h, k) \geq \text{sim}(d_j, k) \quad \text{for } l \leq j \leq h-1 \quad (10)$$

式 (9), 式 (10) を満たす解は一般に複数存在するため、 k_1, \dots, k_{p+m} に対する別の線形制約式として

$$k_1 + \dots + k_{p+m} = k_1^{(i-1)} + \dots + k_{p+m}^{(i-1)} \quad (11)$$

を考える。ここで $k^{(i-1)} = (k_1^{(i-1)}, \dots, k_{p+m}^{(i-1)})$ は前回のクエリーベクトルである。以上の制約式 (9), (10), (11) を満たし、かつ、 $k^{(i-1)}$ と k の類似度が最大になるようなベクトル k を線形計画法¹²⁾ によって求め、新しいクエリーベクトル $k^{(i)}$ とする。ただし、そのような解 k が存在しない場合は $k^{(i)} = k^{(i-1)}$ とする。

4. 被験者を用いた実験

提案手法によってランキングを調整することで検索結果の質が向上するかどうかを調べるため、提案手法を実装した試作システムを用いて以下の評価実験を行った。まず6名の被験者を用いた実験を行って提案手法の有効性を評価した (本章)。次に、本章の実験結果をもとに3種類のクエリーベクトル推測手法を改良したうえで、各手法をより詳しく比較検討しベンチマークを獲得するための実験を行った (5章)。

本章で述べる実験では、被験者に検索課題とその初期ランキングを与え、任意回数調整操作を行ってもらう。実験結果の評価は以下に注目して行った。

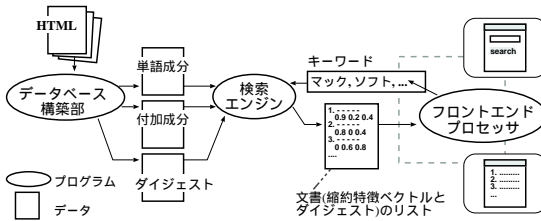


図 2 試作システム
Fig. 2 Prototype system.

- 調整によってより多くの適合文書を上位に配置することができたか。
- 調整によって適合文書の順位を選択的に上げることができたか。すなわち、適合文書の順位は上昇し、不適当な文書の順位は下降しているか。
- ユーザの調整意図をうまく推測し、新しいクエリベクトルに反映させることができたか。また、ユーザの調整意図を正しく推測できた場合とそうでない場合で結果にどのような影響があったか。

4.1 試作システムの概要

試作システムは、データベース構築部、検索エンジン、フロントエンドプロセッサからなる(図2)。データベース構築部と検索エンジンは、既存のキーワード型検索システムである *Freya*⁴⁾ を拡張して実装した。これらはフロントエンドプロセッサからキーワードを入力として受け取り、以下を出力する。

- 指定されたキーワードに関連する文書の集合
- 上記文書集合中の各文書に対する縮約特徴ベクトルとダイジェスト文

フロントエンドプロセッサはユーザの入力したキーワードを検索エンジンに渡し、検索エンジンから返された情報と現在のクエリベクトルを用いてランキングを作成・表示する。検索エンジンから返された情報はキーワードが変更されない限り保持しておき、ユーザがランキングの調整操作を行った場合には再度検索エンジン側と通信することなくランキングの変更を行う。フロントエンドプロセッサには3章で述べた各クエリベクトル推測手法をそれぞれシステム A, B, Cとして実装し、実験に用いた。

- システム A: 成分補正法(3.1節)
- システム B: E 尺度に基づく手法(3.2節)
- システム C: 順序情報に基づく手法(3.3節)

縮約特徴ベクトル中、キーワードとの関連度を表す成分 ($c_{kw_u, d}$) は、 $tf * idf$ に基づき、*Freya* で用いら

表 1 実験に使用した検索課題とキーワード

Table 1 Topics and keywords used in the experiments.

番号	課題名	適合文書の数*1
i	マック用ソフトウェアに関するリンク集 マック, ソフト, ダウンロード, リンク	36
ii	2001 年に国内で開催される国際会議 国際, 会議, 2001	33
iii	大阪府内で最近国際会議に使用された会場 国際会議, 大阪, 会場	13
iv	奈良先端大周辺のラーメン屋 ラーメン, 大阪, 奈良	43
v	禁煙を成功させるための工夫 禁煙, 工夫	48

*1: あらかじめ実験者が設定した正解文書の数

れている以下の定義に従った。

$$c_{kw_u, d} = tf_{kw_u, d} \cdot idf_{kw_u} / rlen_{kw_u, d}.$$

ただし、 $tf_{kw_u, d}$ は d 中での kw_u の出現回数(標頭や見出しは重み付けされている)、 $idf_{kw_u} = \log(2|D_{all}|/|D_{kw_u}|)$ は文書集合全体での kw_u の出現回数の少なさを表す値(D_{all} はデータベース中の全文書、 D_{kw_u} は kw_u を含む文書の集合)、 $rlen_{kw_u, d}$ は d の長さ D_{kw_u} の要素の平均長との比である。また単語以外の文書の特徴を示す成分 ($c_{fv, d}$) として、本章の実験では“文書中のリンク密度”、“更新日の新しさ”の2成分を用いた。ただし、リンク密度とは、 d 中のリンクの数を、 d の長さ d と平均文書長との比で割ったものである。これらの成分を含めることで、4.2節で述べる課題 i, iii のようにリンク集となっているような文書やなるべく新しい文書を探したいとき、うまくクエリベクトルを決めればそのような文書に高いスコアを与えることができる。

各成分間の値の不均衡を防ぐため、検索エンジンから調整対象文書集合 D を受け取った時点で、フロントエンドプロセッサにおいて、第 i 成分の各値を D における第 i 成分の最大値で割る(すなわち、補正後、各成分の最大値が1になるようにする)という補正を行った。

式(4)中の α は両実験とも 0.25 に設定した。

4.2 実験環境

検索課題

検索課題として表1にあげた5課題を設定した。今回の実験では、2.1節手順(1)で入力するキーワードはあらかじめ実験者が指定したものをを用いる(表1各課題の2行目)。被験者に対しては各課題の説明文(100字程度)とキーワードを提示し、課題の内容について十分理解したことを確認のうえ、実験を行った。これ

今回の実験では、システム A, B, C とも計算時間は 100 文書の再ランキングに対して 1 秒程度である。

ら 5 つの課題は、クエリーベクトルの各成分の中で特に入力キーワードに対する重みがランキングの適切さに影響すると思われる課題 (課題 ii, iv, v) と、“文書中のリンク密度” や “更新日の新しさ” など単語以外の成分に対する重みが影響すると思われる課題 (課題 i, iii) からなるようにという意図で決定した。

文書集合

文書データベースは、課題ごとに表 1 のキーワードを既存の WWW 検索サービスに入力し、上位から 1,000 文書分の URL を収集して作成した。実験では、各課題について、上記データベースと表 1 のキーワードを試作システムの検索エンジンに入力した場合の上位 100 文書までを操作対象文書集合とした。

また、各課題の操作対象文書 (100 文書) すべてに対し、課題に適合する (○) か不適合 (×) かを実験者が設定した (被験者には非通知)。

被験者

情報科学研究科の大学院生 6 人で、いずれも既存の WWW 検索サービスを日常的に使用している。

手順

各被験者ごとに以下の手順で実験を行った。

- (1) 実験者が、実験すべき (課題, システム) の組合せを指示する。1 人の被験者は 5 つの課題をちょうど 1 回ずつ、A, B, C のいずれかのシステムを用いて行うよう指示される。たとえば被験者 1 は (v, A), (i, A), (iv, C), (iii, B), (ii, B) の順に実験を行う。各課題ごとにまず実験者がキーワードを入力し、初期ランキングを表示して、被験者に課題の説明を行う。
- (2) 被験者は各課題ごとに以下の作業を 10 分間繰り返し行い、初期ランキングをより適切なものに改良していく。ただし被験者が現在のランキングに満足した場合は、10 分未満でも作業を終了する。
 - (a) 現在のランキング中で調整を行いたい (順位を上げ/下げたい) と感じる文書を探し、調整によってどのような変化を望むのか (調整意図と呼ぶ) を用紙に記述する。調整意図は、クエリーベクトルの各成分に対し、その成分の重要度を「上げたい/下げたい/今回の調整には関係ない」の 3 択形式で記述する。
 - (b) 実際の調整操作を行う。
 - (c) 調整の結果表示されたランキングがあまりにも調整意図から離れていると感じた場合は、『もどる』ボタンによって操作を取り消すことがで

きる。

- (3) 5 課題すべてに対する作業が終了した後、被験者はシステムの使いやすさなどに関する事後アンケートに答える。

被験者には実験前日までに実験の目的やシステムの操作方法をまとめたドキュメントを配布し読んでおいてもらった。また、実験当日にはまず練習用の課題を用いて実験と同じ手順で練習を行い、システムの使用方法について問題がないことを実験者が確認した後に実験を行った。被験者や課題の特性による影響を避けるため、(課題, システム) の組合せ系列、課題の順序、システムの使用順序は全員異なるように設定した。

4.3 測定値

各課題ごとに、操作対象文書の集合を D ($|D| = 100$)、適合文書の集合を D_{rel} ($D_{rel} \subset D$) とし、全体適合率 P を以下のように定義する：

$$\text{全体適合率 } P = \frac{|D_{rel}|}{|D|} = \frac{|D_{rel}|}{100}.$$

また各課題に対する作業ごとに、被験者が i 回目の調整操作を行った後に表示されるランキングにおいて上位 20 位までの文書の集合を $D_{(\leq 20)}^i$ とする。同様に、被験者が i 回目の調整操作を行った場合に、その調整によって新しく上位 20 位以内にランクインした文書の集合を D_{new}^i 、その調整によって順位が上がった文書の集合を D_{up}^i 、順位が下がった文書の集合を D_{down}^i とする。すなわち、第 i 回目調整後のランキングにおける文書 d の順位を $Rank(d, i)$ とすると、

$$\begin{aligned} D_{(\leq 20)}^i &= \{d : Rank(d, i) \leq 20\}, \\ D_{new}^i &= \{d : Rank(d, i-1) > 20 \text{ かつ} \\ &\quad Rank(d, i) \leq 20\}, \\ D_{up}^i &= \{d : Rank(d, i) < Rank(d, i-1)\}, \\ D_{down}^i &= \{d : Rank(d, i) > Rank(d, i-1)\}. \end{aligned}$$

各課題・各被験者の作業に対し、各 i 回目の調整ごとに以下の 4 種類の適合率を定義する：

$$\begin{aligned} \text{20 位以内適合率} \quad P_{(\leq 20)}^i &= \frac{|D_{(\leq 20)}^i \cap D_{rel}|}{|D_{(\leq 20)}^i|}, \\ \text{新規 20 位以内適合率} \quad P_{new}^i &= \frac{|D_{new}^i \cap D_{rel}|}{|D_{new}^i|}, \\ \text{up 適合率} \quad P_{up}^i &= \frac{|D_{up}^i \cap D_{rel}|}{|D_{up}^i|}, \\ \text{down 適合率} \quad P_{down}^i &= \frac{|D_{down}^i \cap D_{rel}|}{|D_{down}^i|}. \end{aligned}$$

ただし分母が 0 である適合率 (たとえば $|D_{down}^i| = 0$)

キーワード型検索システム Freya と同等 .4.1 節で述べた $c_{kw_{u,d}}$ の定義を用いる。

結果表示の 1 ページ目 . 既存の WWW 検索サービスのほとんどが 10~20 文書を 1 ページにまとめて表示していることから、試作システムも 20 文書ずつ表示するよう設計した。

表 2 被験者を用いた試用実験の結果

Table 2 Results of the experiment with human subjects.

	(a) $\frac{P_{\leq 20}^i}{P_{\leq 20}^0}$		(b) $\frac{P_{\text{new}}^i}{P}$		(c) $\frac{P_{\text{up}}^i}{P_{\text{down}}^i}$	
	平均	> 1 *	平均	> 1 *	平均	> 1 *
A	0.86	N(1%)	1.08		1.19	Y(1%)
B	0.83	N(1%)	0.97		1.04	
C	0.97		1.33	Y(1%)	1.15	
課題 i	0.92	N(5%)	1.39	Y(1%)	1.04	
課題 ii	0.88	N(1%)	0.96		1.02	
課題 iii	0.79	N(1%)	1.04		0.99	
課題 iv	1.08	Y(1%)	0.92		1.06	
課題 v	0.78	N(1%)	1.32	Y(1%)	1.41	Y(1%)
被験者 1	1.00		1.33	Y(1%)	1.06	
被験者 2	0.89	N(5%)	1.27		1.17	
被験者 3	0.82	N(1%)	0.92		1.23	
被験者 4	0.86	N(1%)	0.69		1.13	
被験者 5	0.91	N(1%)	1.24		1.07	
被験者 6	0.76	N(1%)	1.04		1.20	
全体	0.89	N(1%)	1.15	Y(1%)	1.13	Y(1%)

*: 平均値が 1.0 よりも有意に高い場合を Y, 低い場合を N, 有意差が認められない場合を空欄で示す. かつこ内は有意水準.

のときの P_{down}^i) は, 以降での分析対象としない.

4.4 結果と考察

(a) 20 位以内適合率

20 位以内適合率が調整前(初期ランキング)よりも高くなっていけば, 調整によってランキングをより適切なものに改良できたと考えられる. そこで, 各 i 回目調整後の 20 位以内適合率と初期ランキングにおける 20 位以内適合率との比率 $P_{\leq 20}^i / P_{\leq 20}^0$ を全課題・被験者について求め, 平均値が 1.0 より高いかどうかの検定を行った. 各システム・課題・被験者ごとの平均値と検定結果を表 2 の (a) に示す.

全体の結果(表の最下行)を見ると, $P_{\leq 20}^i / P_{\leq 20}^0$ の平均値は 1.0 よりも有意に低い値となっており, 初期ランキングよりも 20 位以内適合率は悪くなっているという結果になった. 分散分析の結果 A, B, C のシステム間には有意差が認められ(5%有意), 平均値はシステム C が最も高いが, 1.0 よりも高いとはいえなかった. 課題間・被験者間にも有意差がみられ, 課題 iv のみ平均値が 1.0 より高かった.

(b) 新規 20 位以内適合率

たとえ上位 20 位の適合率が同じでも, 調整によって新しく未知の適合文書(調整前には 20 位以内に入っていなかった適合文書)を多く見つけることができれば, ユーザにとって有益であると考えられる. そこで, 調整によって適合文書を選択的に上位 20 位以内に上昇させることができたかどうかを調べるため, 各 i 回目調整後の新規 20 位以内適合率と全体適合率との比

率 P_{new}^i / P を全課題・被験者について求め, 平均値が 1.0 より高いかどうかの検定を行った(表 2(b)).

全体の結果を見ると, P_{new}^i / P の平均値は有意水準 1% で 1.0 よりも高い値になっており, 調整によって下位に埋もれていた適合文書を選択的に上位に移動することができたといえる. 分散分析の結果システム間には有意差は認められなかったが, システム C のみ平均値が 1.0 よりも有意に高かった. また課題間・被験者間にも有意差は認められなかったが, 課題 i, v と被験者 1 が他に比べ良い結果になった.

(c) up 適合率 vs. down 適合率

上位 20 文書に限らず, 全体として適合文書の順位を上げ, 不適合文書の順位を下げることでいければ, 適切なクエリーベクトル変更が行えたと考えられる. そこで, 調整によって順位の上がった文書の適合率が, 順位の下がった文書の適合率よりも高いかどうかを調べる. 各 i 回目調整後の up 適合率と down 適合率との比率 $P_{\text{up}}^i / P_{\text{down}}^i$ を全課題・被験者について求め, 平均値が 1.0 より高いかどうかの検定を行った(表 2(c)).

全体の結果を見ると, $P_{\text{up}}^i / P_{\text{down}}^i$ の平均値は有意水準 1% で 1.0 よりも高い値になっており, 調整によって起こったランキング(クエリーベクトル)の変化は適切であったといえる. 分散分析の結果システム間には有意差は認められなかったが, システム A のみ平均値が 1.0 よりも有意に高かった. また課題間・被験者間にも有意差は認められなかったが, 課題 v が他に比べ良い結果になった.

(d) 意図どおりの調整ができたか

システムが実際に行ったクエリーベクトル変更が, 被験者が実験手順(2-a)で記述した調整意図(クエリーベクトルの各成分に対する重要度上げ/下げ/無関係)に沿っていたかどうかを調べるため, 以下で定義する値 F, A を考える. 被験者 1 人が 5 つの課題に対して行う全調整操作において,

$F(\text{Follow})$: $P_F(i, j)$ が真となる組 (i, j) の総数

$A(\text{Against})$: $P_A(i, j)$ が真となる組 (i, j) の総数

と定義する. ここで $P_F(i, j)$ は, 被験者が第 i 回目の操作において, 実験手順(2-a)でクエリーベクトルの第 j 成分の値を大きく(小さく)したいと回答し, かつ $k_j^{(i)} > k_j^{(i-1)}$ ($k_j^{(i)} < k_j^{(i-1)}$)であった場合に真となる述語である. すなわち, F とは, 1 人の被験者が行う調整操作全体を通じて, 被験者の意図どおりにクエリーベクトルが変更された回数を意味する. 一方 $P_A(i, j)$ は, 被験者が第 i 回目の操作において, 実験手順(2-a)

表 3 意図どおりの調整ができたか

Table 3 Did the system recognize user's intention?

	F	A*
A	30	17
B	24	11
C	16	46 (17)
課題 i	14	9
課題 ii	9	16 (10)
課題 iii	7	22 (7)
課題 iv	23	12 (10)
課題 v	17	15 (9)
被験者 1	19	10 (9)
被験者 2	8	13 (8)
被験者 3	6	3
被験者 4	8	27 (8)
被験者 5	14	16 (14)
被験者 6	15	5 (3)
全体	70	74 (45)

*: かつこ内は、システム C で「解なし」になった場合を除いた値。

でクエリーベクトルの第 j 成分の値を大きく(小さく)したいと回答し、かつ $k_j^{(i)} \leq k_j^{(i-1)}$ ($k_j^{(i)} \geq k_j^{(i-1)}$)であった場合に真となる述語である。すなわち、 A とは、被験者の意図と逆方向にクエリーベクトルが変更された回数である。表 3 に各システム・課題・被験者ごとの F , A の値を示す。

表 3 より、(b) または (c) の結果において平均値が 1.0 よりも有意に高かった課題 i, v, 被験者 1 は F の値が大きい課題・被験者であったことが分かり、クエリーベクトルをユーザの調整意図に沿って適切に変更することができれば、適合文書を選択的に上位に移動することができると思われる。

一方、システム C は (b) の結果において平均値が 1.0 よりも有意に高い結果が得られたが、 F の値はシステム A, B に比べて小さく、逆に A の値が大きい。順序情報に基づく推測手法では、その定義より、連立不等式の解が存在しない場合が生じる。3.3 節で述べたように、今回の実験システムでは、 i 回目の調整で「解なし」となった場合は $k^{(i)} = k^{(i-1)}$ とした。このとき任意の j について $k_j^{(i)} = k_j^{(i-1)}$ であるため、 A の値が大きくなったと考えられる。「解なし」は今回の実験においておよそ 3 割弱の割合で生じており、前回と同じクエリーベクトルを用いる方法では問題があると思われたため、5 章に述べるベンチマークテストでは「解なし」の場合でもある程度調整意図に沿った変更ができるよう改良を行った(5.1 節)。

(e) アンケート結果より

システムの使いやすさを 10 段階で評価してもらった(評価値が大きいほど被験者が「使いやすい」と感

じたことを意味する)。その結果、全被験者の回答の平均値は 6.6 であった。また、システム A, B, C の違いについて尋ねたところ、6 人中 3 人が「特に違いを感じなかった」と回答した。これは、システムの違いがあまり顕著ではないと同時に、システムによって検索した課題が異なるために比較が難しかったためと考えられる。しかし残り 3 人からは「A はランキングが少ししか変化しなかった」「B は変化しすぎて使いにくかった」「C はまったく変化しないことがあって不便」という回答が得られた。この指摘をもとに、5 章に述べるベンチマークテストではそれぞれの推測手法の改良を目指したパラメータ調整を行った(5.1 節)。

考 察

上記 (a) ~ (d) の結果から、以下の考察が得られる:

- 提案手法を用いることにより、上位 20 文書の適合率を上げることはできないが、下位に埋もれた適合文書を選択的に上位に持ち上げることが可能であり、未知の適合文書を発見しやすくなるという点でユーザにとって有効であると考えられる。

- 全体として適合文書の順位を上げ、不適合文書の順位を下げているため、クエリーベクトルを適切に変更できたと考えられる。

- ユーザの調整意図をうまく推測しクエリーベクトルに反映できた場合は良い結果が得られる。しかし調整意図をうまく推測できない場合が半分程度生じており、各手法の改良が必要である。

5. ベンチマークテスト

ベンチマークテストでは、被験者の代わりにある調整戦略に従ってランキングの調整を行った場合の各種適合率を求める。同時に、各クエリーベクトル推測手法 A, B, C の性能を、4 章の実験結果をもとに改良・調整を行ったうえでより詳細に比較することが目的である。実験結果の評価は 4 章と同じく、20 位以内適合率・新規 20 位以内適合率・up および down 適合率によって行い、調整によってランキングを適切に変更できたか、4 章の実験結果と違いはあるか、より良いシステム(推測手法)とはどのようなものか、などに注目して考察を行う。

5.1 実験システム

4 章の実験結果から得られた指針をもとに、クエリーベクトルの推測手法 A, B, C をそれぞれ以下のように改良・調整し、実験に用いた。

システム A: 成分補正法

事後アンケートで「(調整しても)変化が小さい」という指摘があった。この原因として、ベクトル md_i

を作成する際の補正が効きすぎている（小さすぎる）のではないかと考え、補正值 $\eta_j \in \{0, 0.5, 1.0\}$ を以下の3通りに変更しそれぞれシステム A1, A2, A3 とした：

- システム A1 : $\eta_j \in \{0.1, 0.5, 1.0\}$
- システム A2 : $\eta_j \in \{0.25, 0.75, 1.0\}$
- システム A3 : $\eta_j \in \{0.5, 0.75, 1.0\}$

システム B : E 尺度に基づく推測手法

事後アンケートで「変化が大きすぎる」という指摘があった。この原因として、他の2手法では新しいクエリベクトル $k^{(i)}$ を求める計算中に前回のクエリベクトル $k^{(i-1)}$ が用いられているのに対し、 E 尺度に基づく手法では $k^{(i-1)}$ をまったく使用しないということが考えられる。そこで、3.2 節の式 (8) によって k を求めた後に前回のクエリベクトル $k^{(i-1)}$ との重み付き平均をとり、これを $k^{(i)}$ とする改良を行った。すなわち

$$k^{(i)} = \xi k + (1 - \xi) k^{(i-1)} \quad (12)$$

とし、 ξ を 0.25, 0.5, 0.75, 1.0 の4通りに設定してそれぞれシステム B1, B2, B3, B4 とした。システム B4 は $k^{(i-1)}$ に対する重みが 0 になるため、従来のシステム B と同等である。

システム C : 順序情報に基づく推測手法

「解なし」が高頻度に発生し、事後アンケートで「まったく変化しないのは不便」という指摘があったため、「解なし」の場合に以下の方法で新しいクエリベクトルを作成するよう改良を行った。

(1) 3.3 節の式 (10) の代わりに次式 (13) を用いて再度計算を行う。ただし d_{lows} は $d_l, d_{l+1}, \dots, d_{h-1}$ の平均ベクトル (3.1 節)。

$$\text{sim}(d_h, k) \geq \text{sim}(d_{lows}, k) \quad (13)$$

(2) 上記手順 (1) でも「解なし」になった場合、 E 尺度に基づく手法 (3.2 節) で定義したベクトル $g(\text{good})$ (d_h と d_{l-1} の平均ベクトル) を求め、新しいクエリベクトル $k^{(i)}$ とする。これは、システム C で前回と異なるクエリベクトルを作成できない場合、システム B を簡略化して用いることを意味する。

上位固定と上位変化

事後アンケートにて「調整の後、前のランキングで上位にあった文書をどこか(下位)へ見失ってしまい、探すのが大変だった」という指摘があった。このことから、ユーザが文書 d_h を文書 d_l ($1 \leq l < h \leq |D|$) のすぐ上に移動する調整操作を行った場合、文書 d_1 から d_{l-1} は調整の前後で同順位(調整によって順位が変化しない)とした方が利便性が高いと考えられる。

そこで、先に述べた8システム(A1~A3, B1~B4, C)すべてに対し、文書 d_1 から d_{l-1} まではスコアの再計算を行わず前回と同じ順位で出力するもの(上位固定版)を用意し、上位も変化する従来のシステム(上位変化版)と比較を行った。上位固定版と上位変化版は、2.1 節手順 (4) までは完全に同じ動作を行う。

ベンチマークテストは以上の16システム(上位固定版 A1~C, 上位変化版 A1~C)に対して行った。

このほか、4.1 節で述べた縮約特徴ベクトルの成分間の不均衡を防ぐ補正として、第 i 成分の各値を調整対象文書集合 D における第 i 成分の平均値で割る(すなわち、補正後、各成分の平均値が1になるようにする)よう変更した。また、初期クエリベクトルを $k^{(0)} = (\underbrace{1, 1, \dots, 1}_p, \underbrace{0.5, 0.5, \dots, 0.5}_m)$ とするよう変更した。

5.2 実験環境など

検索課題と文書集合

検索課題には、4 章の被験者実験と共通する4課題(表1の課題 i, ii, iv, v)を用いた。各課題に対する文書集合と適合文書集合は4章の実験と同じである。ただし、精度をより厳密に議論するため、課題ごとの適合文書()をさらに“優”と“良”に分類した。ここで優文書とは、検索課題に対して非常によく適合している文書であり、被験者の違いによって適合・不適合の判断が左右されないと考えられるものである。後の分析では D_{rel} の代わりに優文書の集合 $D_{優}$ を用いて各種適合率を求めた。

手 順

各課題ごとに以下の作業を機械的に行った。

- (1) 現在のランキングを上位から順にチェックし、不適合文書(x)が2つ以上連続している箇所を見つける。
- (2) その下に現れる最初の適合文書 d_h ()を、連続したxのすぐ上まで持ち上げる。

ユーザが調整を行いたいと感じるのは、システムの考える適合基準と自分自身の適合基準とが明らかに異なると感じるときであると考え、そのような状況を「xの連続」と定義した。

測 定 値

測定値は4章の実験と同様に、20位以内・新規20位以内・up・downの4種類の適合率である。上位固定版のシステムにおいては、調整によってスコアを再計算した文書のみを対象として各適合率を求めた。す

課題 iii には、この段落で述べる「優文書」がなかったため、ベンチマークテストでは除外した。

なわち, i 回目の調整で文書 d_h を文書 d_l のすぐ上に移動した場合, 1 位から $l-1$ 位までの文書についてはスコア再計算を行わないため調整による変化が生じない. よって新しいランキングでは第 l 位を 1 位と考へ, 20 以内文書集合 $D_{(\leq 20)}^i$ と新規 20 位以内文書集合 D_{new}^i の定義を以下のように変更した.

$$D_{(\leq 20)}^i = \{d : \text{Rank}(d, i) \leq l + 19\},$$

$$D_{new}^i = \{d : \text{Rank}(d, i - 1) > l + 19 \text{ かつ} \\ \text{Rank}(d, i) \leq l + 19\}.$$

5.3 結果と考察

(a) 20 位以内適合率

各 i 回目調整後の 20 位以内適合率と初期ランキングにおける 20 位以内適合率との比率 $P_{(\leq 20)}^i / P_{(\leq 20)}^0$ を全課題について求め, 平均値が 1.0 より高いかどうかの検定を行った (表 4 (a)). 全体の結果 (表の最下行) を見ると, $P_{(\leq 20)}^i / P_{(\leq 20)}^0$ の平均値は 1.0 よりも有意に低い値となっており, 初期ランキングよりも 20 位以内適合率は悪くなっている. 上位固定版と上位変化版の間には有意差が認められ (1%有意), 上位変化の方が上位固定に比べ平均値が高く, また 1.0 よりも有意に高い値となった. A1~C の 8 システム間に有意差は認められず, また平均値が 1.0 よりも高いシステムはなかった.

(b) 新規 20 位以内適合率

各 i 回目調整後の新規 20 位以内適合率と全体適合率との比率 P_{new}^i / P を全課題について求め, 平均値が 1.0 より高いかどうかの検定を行った (表 4 (b)). 全体の結果を見ると, P_{new}^i / P の平均値は有意水準 1% で 1.0 よりも高い. つまり, 調整によって下位に埋もれていた適合文書を選択的に上位に移動することができたといえる. 上位固定と上位変化の間には有意差は認められなかったが, 上位固定のみ平均値が 1.0 よりも有意に高かった. システム間にも有意差は認められなかったが, システム A1, A3 は平均値が 1.0 よりも有意に高く, その他の 6 システムについては 1.0 よりも高いとはいえなかった.

(c) up 適合率 vs. down 適合率

各 i 回目調整後の up 適合率と down 適合率との比率 P_{up}^i / P_{down}^i を全課題について求め, 平均値が 1.0 より高いかどうかの検定を行った (表 4 (c)).

全体の結果を見ると, P_{up}^i / P_{down}^i の平均値は有意水準 1% で 1.0 よりも高い値になっており, 調整によって起こったランキング (クエリーベクトル) の変化は適切であったといえる. 上位固定と上位変化の間には有意差は認められず, どちらの結果も平均値は 1.0 より

表 4 ベンチマークテストの結果

Table 4 Results of the benchmark test.

	(a) $\frac{P_{(\leq 20)}^i}{P_{(\leq 20)}^0}$		(b) $\frac{P_{new}^i}{P}$		(c) $\frac{P_{up}^i}{P_{down}^i}$	
	平均	> 1 *	平均	> 1 *	平均	> 1 *
固定	0.63	N(1%)	1.23	Y(5%)	1.54	Y(1%)
変化	1.18	Y(1%)	1.12		1.35	Y(1%)
A1	0.99		1.40	Y(1%)	1.63	Y(5%)
A2	0.87	N(5%)	1.30		1.14	
A3	0.93		1.32	Y(5%)	1.10	
B1	0.88	N(1%)	1.00		1.35	Y(5%)
B2	0.87	N(1%)	1.16		1.68	Y(1%)
B3	0.92		1.12		1.37	
B4	0.91		0.76		1.62	Y(1%)
C	0.89		1.31		1.59	Y(1%)
全体	0.91	N(1%)	1.17	Y(1%)	1.44	Y(1%)

*: 平均値が 1.0 よりも有意に高い場合を Y, 低い場合を N, 有意差が認められない場合を空欄で示す. かつこ内は有意水準.

も有意に高かった. システム間にも有意差は認められなかったが, システム A1, B1, B2, B4, C について平均値が 1.0 よりも有意に高いといえた.

考 察

上記 (a) ~ (c) の結果から, 以下の考察が得られる:

- 全体の傾向として, 調整によって上位 20 文書の適合率を上げることはできないが, 下位に埋もれた適合文書を選択的に上位に持ち上げることが可能であり, 未知の適合文書を上位に発見できるという点で有効な手法といえる.
- 上位固定と上位変化を比較すると, 上位変化の方が上位固定を若干上回る結果となった. ただし, ユーザの感じる使いやすさ, 負担の少なさ, という観点からの評価も必要であり, 双方を使い分けるインタフェースなども考慮すべきと思われる.
- システム間の比較については, 今回の実験では (a), (b), (c) いずれの結果においても有意差は認められなかった. ただし問合せベクトル変更の適切さを表す (c) の結果においてシステム B2, A1, B3, C の順に平均値が高いことから, システム A, B, C で比較すると $B > A > C$ という傾向が窺える. パラメータの違いによる比較では, システム A の 3 種類中では A1, システム B の 4 種類中では B2 が最も良い結果になった.
- 10 回の調整を比較すると, すべてのデータに共通して, 1, 2 回目の調整後に最大値となる傾向が見られた. このことから, 調整によるランキング改良はただか 2 回程度の操作で最も効率良く行えることが分かり, ユーザにとって無理のない方法と考えられる.

6. ま と め

WWW 検索作業におけるユーザの心理的負担の軽減を目的に、文書順位の上げ/下げ操作によってユーザが対話的にランキングを調整できる手法を提案し、実験によってその有効性を検証した。

被験者を用いた実験の結果から、提案手法によって上位文書の適合率を上げることはできないが、下位に埋もれた適合文書を選択的に上位に持ち上げることが可能であり、未知の適合文書を発見しやすいという点で有効な手法であると確認できた。また、全体として適合文書の順位を上げ、不適合文書の順位を下げており、提案した推測手法によってクエリーベクトルを適切に変更できたと考えられるが、特にユーザの調整意図をうまく推測できた場合にはその効果が大きく、推測手法の精度向上が重要であることが分かった。

ベンチマークテストでは被験者を用いた実験から得られた知見をもとに 3 推測手法それぞれを改良し、より厳密に比較評価を行った。その結果、被験者を用いた実験と同じ傾向として、下位に埋もれた適合文書を選択的に上位に持ち上げていること、全体としてクエリーベクトルを適切に変更できていることを確認した。また、調整によるランキング改良手法はたかだか 2 程度の操作で最も効率良く行えることが分かり、アンケートの結果からも、提案手法がユーザにとって無理のない方法であることが確認できた。手法間の比較では特にシステム B4, C が他に比べ優れていることが分かった。一方、調整操作が行われた部分よりも上位の文書の順位を固定する方法は、従来の(固定しない)方法に比べて良いとはいえない結果であったが、今後被験者による評価によって再検討する必要があると感じられた。

今後、ユーザの調整意図をより正確に推測するための改良を続けると同時に、被験者を用いたさらなる評価実験を行う予定である。特に、従来の適合フィードバック手法や統計的分類手法との比較を、性能・使いやすさの両面から行いたいと考えている。文献 8) は対話的情報検索システムの評価実験手法について述べたもので、参考にすべきところが多い。たとえば、本論文 4 章では、実験結果に対する分散分析によって手法間に有意差があるかどうか調べたが、この方法では差の大きさの見積りはできなかった。一方、文献 8) では、実験計画の工夫により、システム間の性能の差

を見積もれるようにしている。また、本手法は、調整操作という手間をかける代わりに、検索精度の向上やその他の利益を得ようとするものであるため、これらの手間と利益との比に着目した評価を行うべきと考えられる。そのような評価方法についても考察が必要である。

ユーザの調整意図を正確に推測するための改良としては、縮約特徴ベクトルの成分の追加を検討している。現在はユーザの入力したキーワード、リンク密度、更新の新しさをを用いているが、今後はキーワードの類義語やキーワードとの共起性の高い語なども加えることで推測精度の向上を目指す予定である。

謝辞 試作システムの実装には、検索システム『Freya』を利用させていただきました。この場を借りて作者の方に深く敬意を表します。

参 考 文 献

- 1) Anick, P.G. and Tipirneni, S.: The Paraphrase Search Assistant: Terminological Feedback for Iterative Information Seeking, *Proc. SIGIR 99*, pp.153-159 (1999).
- 2) Bruza, P., McArthur, R. and Dennis, S.: Interactive Internet Search: Keyword, Directory and Query Reformulation Mechanisms Compared, *Proc. SIGIR 2000*, pp.280-287 (2000).
- 3) 江口浩二, 伊藤秀隆, 隈元 昭, 金田彌吉: 漸次的に拡張されたクエリーを用いた適応的文書クラスタリング法, 電子情報通信学会論文誌, Vol.J82-D-I, No.1, pp.140-149 (1999).
- 4) 原田昌紀: *Freya version 0.92* (1998).
<http://odin.ingrid.org/freya/>
- 5) 河野浩之, 長谷川利治: WWW データ資源に対する重み付き相関ルール導出アルゴリズムの適用, 重点領域研究「高度データベース」松江ワークショップ講演論文集, Vol.1, pp.90-99 (1996).
- 6) 木下敦史, 仲川こころ, 高田喜朗, 関 浩之: 対話的に文書ランキングを調整できる WWW 検索支援手法, 情報処理学会研究報告, 2001-FI-64, pp.63-70 (2001).
- 7) 仲川こころ, 高田喜朗, 関 浩之: 可変なカテゴリ構造を用いた文書検索支援手法, 情報処理学会論文誌, Vol.42, No.10, pp.2441-2453 (2001).
- 8) Over, P.: TREC-7 Interactive Track Report, *The 7th Text REtrieval Conference (TREC 7)*, pp.65-71, National Institute of Standards and Technology (1999).
- 9) Page, L., Brin, S., Motwani, R. and Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web (1998).

ただし、改良後のシステム C は「解なし」の場合にシステム B の手法を簡略化して実行しているため、純粋に A, B との比較はできない。

<http://citeseer.nj.nec.com/345917.html>

- 10) Pirolli, P., Shank, P., Hearst, M. and Diehl, C.: Scatter/Gather Browsing Communicates the Topic Structure of a Very Large Text Collection, *Proc. CHI 96*, pp.213-220 (1996).
- 11) Pollitt, S.A.: The Key Role of Classification and Indexing in View-based Searching, *Proc. 63rd IFLA General Conf.* (1997).
<http://www.ifla.org/IV/ifla63/63cp.htm>
- 12) Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P.: *Numerical Recipes in C*, 2nd edition, pp.430-444, Cambridge University Press (1992).
- 13) Rocchio Jr., J.: Relevance Feedback in Information Retrieval, *The SMART Retrieval System: Experiments in Automatic Document Processing*, Salton, G. (Ed.), pp.313-323, Prentice Hall Inc. (1971).
- 14) Salton, G., Singhal, A., Buckley, C. and Mitra, M.: Automatic Text Decomposition Using Text Segments and Text Themes, *Proc. Hypertext 96*, pp.53-65 (1996).
- 15) Takata, Y., Nakagawa, K. and Seki, H.: Flexible Category Structure for Supporting WWW Retrieval, *Proc. ER2000 Conference Workshop on the World Wide Web and Conceptual Modeling (WCM2000)*, LNCS1921, pp.165-177 (2000).
- 16) van Rijsbergen, C.: *Information Retrieval*, 2nd edition, Butterworths (1979).
- 17) Witten, I.H., Moffat, A. and Bell, T.C.: *Managing Gigabytes: Compressing and Indexing Documents and Images*, Von Nostrand Reinhold, New York (1994).

付 録

3.2 節の式 (7) から式 (8) を導出する手順を説明する。一般性を失うことなく、ベクトル g, b, k は正規化されているものと仮定する (すなわち $|g| = |b| = |k| = 1$)。 g と b のなす角を θ , g と k のなす角を x とすると、式 (7) を満たす k が g と b が作る平面上に存在することは明らかである。したがって b と k のなす角は $x + \theta$ となる (図 3)。

式 (3), (6) より、式 (7) の r は以下のような x の関数として表される。

$$f(x) = \frac{\cos x \sin(x + \theta)}{\cos x + \sin(x + \theta)} \tag{14}$$

式 (14) の $f(x)$ を最小にする x を求めるため $df/dx = 0$ とすると、以下の関係が成り立つ。

$$(\cos(x + \theta) - \sin x)(1 + \sin x \cos(x + \theta)) = 0 \tag{15}$$

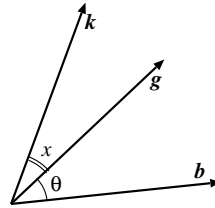


図 3 g, b, k の関係
Fig. 3 Vectors g, b, k .

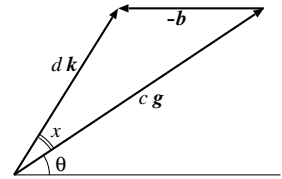


図 4 $dk = cg - b$
Fig. 4 $dk = cg - b$.

式 (15) を満たす解は以下のとおりである。

$$\cos x = \sqrt{\frac{\sin \theta + 1}{2}} \text{ or } x = \frac{1}{2}(\frac{\pi}{2} - \theta) \tag{16}$$

今、ある正の実数 c と d を用いてベクトル k を式 (17) のように表すと、式 (18), (19) が成り立つ (図 4)。

$$dk = cg - b \tag{17}$$

$$c \sin \theta = d \sin(x + \theta) \tag{18}$$

$$c \cos \theta = 1 + d \cos(x + \theta) \tag{19}$$

式 (16), (18), (19) より、

$$c = \frac{1 + \sin \theta}{\cos \theta} \tag{20}$$

g との類似度と b との非類似度の調和平均を用いて k の最適解を求める本手法は、必ずしも文書 d_h をちょうど d_{l-1} と d_l の間に配置できるものではないが、たかだか数回のベクトル正規化と内積だけの計算で済むため高速に実行できるという利点がある。

(平成 13 年 11 月 26 日受付)

(平成 14 年 7 月 2 日採録)



仲川こころ (正会員)

平成 9 年関西学院大学理学部物理学卒業。平成 14 年奈良先端科学技術大学院大学情報科学研究科博士後期課程修了。在学中、情報検索、ヒューマンインタフェースに関する研究に従事。現在、通信総合研究所けいはんな情報通信融合研究センターに勤務。ひと同士の柔軟なコミュニケーションを可能にする、認知や発達の仕組みに興味を持つ。博士 (工学)。情報処理学会第 63 回大会奨励賞。



木下 敦史

昭和 52 年生。平成 12 年大阪教育大学教育学部情報科学専攻卒業。平成 14 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了。修士(工学)。在学中、情報検索を対象としてシステムの使いやすさに関する研究に従事。現在は三洋電機株式会社デジタルシステム研究所勤務。



高田 喜朗(正会員)

平成 4 年大阪大学基礎工学部情報工学科卒業。平成 9 年同大学院博士後期課程修了。同年奈良先端科学技術大学院大学情報科学研究科助手。現在に至る。博士(工学)。ユーザインタフェースの仕様記述法、情報検索に関する研究に従事。



関 浩之(正会員)

昭和 62 年大阪大学大学院基礎工学研究科博士後期課程修了。工学博士。同年大阪大学基礎工学部情報工学科助手。同講師、助教授を経て、平成 6 年奈良先端科学技術大学院大学情報科学研究科助教授。平成 8 年同教授、現在に至る。形式言語理論、ソフトウェアの基礎理論に関する研究に従事。平成 9 年度情報処理学会論文賞。