

構造化プロフィールを用いた個人化 Web 検索システム

岩崎 周造[†] 太田 学[†]

[†] 岡山大学大学院自然科学研究科 〒700-8530 岡山市津島中 3-1-1

E-mail: [†] {iwasaki, ohta}@de.it.okayama-u.ac.jp

あらまし Web 検索において、ユーザごとに異なる検索意図に対応するため、個別のランキング結果を提示する手法が個人化 Web 検索である。しかし、同一のユーザでも多様な興味の状態を持ち、その状態によって要求する Web ページも変化する可能性がある。そのため、一様な個人化が全ての興味状態に有効に働くとは限らない。そこで本研究では、それら多様な興味状態を反映するため、ユーザプロフィールを構造化した個人化 Web 検索手法を提案する。

キーワード Web 検索, パーソナライゼーション, 構造化

Personalized Web Search Using a Structured Profile

Shuzo IWASAKI[†] and Manabu OHTA[†]

[†] Graduate School of Natural Science and Technology, Okayama University

3-1-1 Tsushimanaka, Okayama-shi, Okayama, 700-8530 Japan

E-mail: [†] {iwasaki, ohta}@de.it.okayama-u.ac.jp

Abstract This paper proposes a personalized Web search technique giving a personalized rank list of search results. We know a user has a variety of interests and requests different results depending on their varying interests. Therefore, the same personalization does not necessarily work effectively even for the same user. Our personalization technique uses a structured profile in order to handle such various interests of one user.

Keyword Web Search, Personalization, Structuring

1. はじめに

インターネットの利用において、Google[3]やYahoo![5]を始めとした Web 検索は非常に重要なシステムである。このような Web 検索システムでは、データベース化した膨大な Web ページ情報から、独自の手法で検索結果をランキングし提示している。これにより、クエリに対して一般的に有用度が高いであろうページが上位にランクされ、ユーザが要求するページを発見しやすくなる。

しかし、インターネットの普及に伴い、個人ページやブログなど Web 上の情報量の増加が著しい。また、クエリによっては複数の意味を持つ語がある。例えば“マック”という単語は、OS である“マッキントッシュ”とファーストフード店の“マクドナルド”の両方を表わし、検索結果にはどちらのページも含まれてしまう。以上から、増加する雑多なページや複数の意味を持つクエリによるページはユーザの検索意図と異なるノイズとなり、検索時間や手間の増大に繋がる。

その対策として、SNS(ソーシャルネットワークワーキングサービス)検索や個人化検索が挙げられる。SNS 検索では、似た興味を持つユーザのコミュニティーを作成し、

そのコミュニティー内で優良なサイトを推薦しあう。個人化検索では、ユーザの興味情報をプロフィールとしてデータ化し、それに沿った検索結果のランキングをユーザごとに提示する。これらの β サービスとして、My Web[6]や Google Personalized Search[3]が試験的に運営されている。本研究ではランキングを個人化する個人化検索を扱う。

また、同じユーザでも仕事に関心があるときや趣味に関心があるときなど、複数の興味の状態を持ち、その興味状態に応じて、要求する Web ページも異なると仮定する。この場合、一様な個人化がユーザの全ての興味状態に対し有効に働くとは限らない。そこで、本研究では構造化したユーザプロフィールを用い、多様な興味状態に対応できる個人化 Web 検索手法を提案する。構造化したプロフィールは、ユーザの多様な興味状態を表わす。

関連研究として、井上らの“興味傾向単語の抽出によるパーソナライズド検索システムの提案と実装[1]”がある。この研究で井上らは、Google の検索結果から得られた特徴語をそのユーザの興味を表す興味語とし、ランキングの個人化に用いている。本研究では、個人

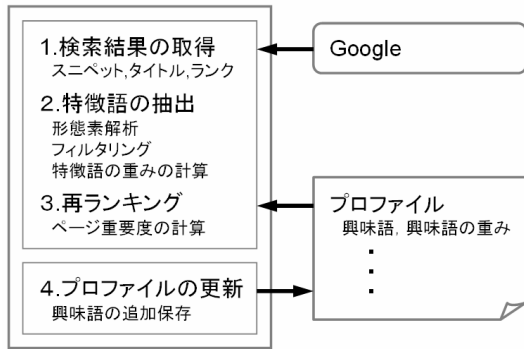


図 1: 個人化処理の流れ

化手法にこの興味語を採用した。また、プロフィールの構造化を行わない一般的な個人化について、“興味単語を用いた個人化 Web 検索[2]”で検証を行い、一定の効果を得た。

本稿では、つづく 2 節と 3 節で提案手法の理論を説明し、4 節でその理論に基づき実装したシステムについて紹介する。5 節では、実装システムを用い実験とその結果の考察を行った。最後に 6 節で本研究をまとめる。

2. 興味単語を用いた検索結果の個人化

本手法では Google の検索結果を形態素解析して得られた特徴語と、ユーザプロフィールに保存された興味語を用いて個人化を行う。興味語とは、特徴語のうちユーザの興味を表す語である。それぞれの語には重みがあり、その重みから各ページの重要度を求め再ランキングに用いる。本節では、まず興味状態を考慮しない個人化手法について説明する。図 1 は、個人化処理の流れ図である。処理 3 が終了した時点でユーザに検索結果を提示し、ユーザがページをクリックすると処理 4 が行われる。

2.1. 特徴語の抽出と重みの計算

特徴語を Google の検索結果から抽出し、その重みを計算する。

形態素解析とフィルタリング

Google の検索結果のうち、タイトルとスニペットについて形態素解析を行い、特徴語を抽出する。形態素解析には MeCab[7]を用いた。MeCab では、文を名詞、動詞、助詞、接続詞などの品詞に分解し、さらに一般名詞、自立名詞、係助詞などに細分類する。本手法では、これらのうち一般名詞、サ変接続名詞、固有名詞を各ページの内容を表すのに相応しい語と考え特徴語とした。例として、一般名詞は海や川、サ変接続名詞は運転や発表、固有名詞は日本やアメリカなどが挙げ

られる。

しかし、これら名詞の中には特徴を表す語として不適切なものもある。そこで、それらを見捨て語として除外するようフィルタリングを行った。無視語は、http, www, web といったインターネット用語や、情報、案内、一覧などの語である。

特徴語の重みの計算

各ページの特徴語について、それぞれの重みを計算する。重みには TF-IDF 法を用いた。しかし、クエリはどのページにも出現するため、クエリに含まれる特徴語の出現ページ数 df は総ページ数と等しく突出して大きい。また、クエリと関連性が強い特徴語の df も大きくなる場合があり、これらの $tfidf$ 値は極端に小さい値となる。この影響を小さくするため、 df の平方根を取り補正を行った。

2.2. ランキングの個人化

特徴語の重みから各ページの重要度を計算し、再ランキングを行う。再ランキングに用いるページの重要度は、特徴語と興味語の重みから計算した個人化ページ重要度と元のランクから求めたランクページ重要度の和で定義される。

個人化ページ重要度

式(1)はページ p の個人化ページ重要度 PI_p の計算式である。個人化ページ重要度 PI_p はページ p に含まれる特徴語 w の重み $tfidf_w$ と興味語 w の重み $weight_w$ から求める。各ページの特徴語数には、ばらつきがあるため、特徴語数 NW_p で割った。

$$PI_p = \frac{\sum_w tfidf_w \cdot weight_w}{NW_p} \quad (1)$$

ランクページ重要度

式(2)はページ p の元のランク r_p を基にしたランクページ重要度 RI_p の計算式である。元のランク r_p は、Google の検索結果での順位であり、 N は取得ページ数である。

$$RI_p = N - r_p + 1 \quad (2)$$

ページ重要度

式(3)はページ p のページ重要度 I_p の計算式であり、個人化ページ重要度とランクページ重要度の和で表される。個人化ページ重要度 PI_p はその最大値 PI_{\max} で

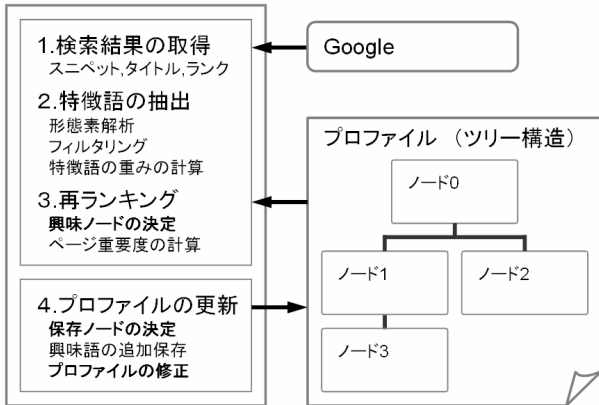


図 2: 構造化プロフィールと個人化処理の流れ

割り, ランクページ重要度 RI_p は取得ページ数 N で割って正規化する. 個人化率 $rate$ は, PI_p と RI_p の比率を表している. この比率を調節することで, 個人化の度合いを変更することが可能である.

$$I_p = \frac{PI_p}{PI_{max}} rate + \frac{RI_p}{N} (1 - rate) \quad (3)$$

こうして得られたページ重要度順にページをソートし, 検索結果の再ランキングを行う.

2.3. プロファイルの更新

ユーザが検索結果のタイトルをクリックし, リンクされている Web ページに移動すると, そのページに含まれる特徴語が興味語としてプロフィールに保存される. 保存されるのは特徴語 w とその重み $tfidf_w$ であり, 興味語 w がプロフィールに存在しない場合は新規作成され, 存在する場合は既存の重みに加算される.

また, 追加保存の前にユーザの興味情報の鮮度を保つための忘却処理を行う. 忘却処理では, 忘却係数 f を定め, プロファイルの更新時に全興味語の重みに乗算する.

3. 多様な興味状態への対応

ユーザの多様な興味状態に対応するため, 本手法ではプロフィールの構造化を行った. 興味状態とは, 検索時のユーザの興味や嗜好の状態である. 本節では, 構造化の詳細と, 2 節で述べた個人化にどのように使用するかを詳述する. 図 2 は, 構造化プロフィールを用いた個人化処理の流れ図である. プロフィールを構造化するにともない, 処理 3 と処理 4 において内部の処理が増えている. これは, 構造化したプロフィールを走査する必要があるためである.

3.1. プロファイルの構造化

本手法で提案するプロフィールはツリー構造をもつ. ツリー構造の各ノードは, ユーザの興味状態を表しており, それぞれ別々に興味語が保存される. 個人化では, ユーザの現在の興味状態に対応するノードを推定し, そのノードと親ノードの興味語を用いてページ重要度を求める.

また, ツリー構造において親ノードはその全子ノードの興味語を全てもつ. ただし, 親ノードの興味語の重みは子ノードに比べ小さくなるようにした. よって, 下位のノードほど専門性が増すこととなる.

3.2. 興味ノードの決定

検索結果には, 現在のユーザの興味に関する情報が含まれていると考えられる. そこで, 検索結果からユーザの興味状態を推定し, 興味ノードを決定する. 興味ノードとは, プロファイルのツリー構造において, ユーザの現在の興味状態に最も近いと考えられるノードである. 2.2 節におけるページ重要度は, 興味ノードとその親ノードの興味語を用いて計算する. これにより各興味状態に応じたページ重要度を求めることができる.

興味ノードスコアの計算

興味ノードの決定には, 式(4)で定義する興味ノードスコアを用いる. ノード n の興味ノードスコア INS_n は, 検索結果から抽出した特徴語 w の出現ページ数 df_w とノード n に保存されている興味語 w の重み $weight_w$ から求めた値である. これは, ユーザの興味状態と各ノードとのスコアとみなすことができる. NW_w はノード n の興味語数であり, スコアの補正を行っている.

$$INS_n = \frac{\sum_w df_w \cdot weight_w}{NW_n} \quad (4)$$

ツリーの走査

ツリーの走査は, ルートノードを最初の暫定興味ノードとし幅優先探索で行う. 暫定興味ノードとその全子ノードについて興味ノードスコアの比較を行い, 最も大きいスコアをもつノードを求める. それが暫定興味ノードの場合, そのノードを興味ノードとし, 子ノードの場合は, その子ノードを暫定興味ノードとして同じ処理を繰り返す. また, 興味ノードスコアには閾値 T_{INS} を設け, 最大興味ノードスコアが閾値以下の場合, 暫定興味ノードを興味ノードとする.

親ノードの興味語の補正

ページ重要度の計算には、興味ノードの興味語が少なく興味情報が不足する場合に備え、興味ノードと親ノードがもつ興味語を用いる。ただし、親ノードの興味語の重みについては影響を小さくするために補正を行う。補正值 M_{IN} ($0 \leq M_{IN} \leq 1$) を定め、親ノードの興味語の重みに乗算する。

3.3. 保存ノードの決定

プロファイルの更新は、ユーザが検索結果のいずれかのページをクリックした時に行われる。このとき、クリックされたページの特徴語は保存ノードとその全親ノードに保存される。保存ノードは、クリックされたページと最も関連性が高いと考えられるノードである。

保存ノードスコアの計算

保存ノードを決定する指標には、保存ノードスコアを用いる。ノード n の保存ノードスコア SNS_n は、クリックしたページに含まれる特徴語 w の重み $tfidf_w$ とノード n に保存されている興味語 w の重み $weight_w$ から得られ、式(5)で定義する。 NW_w はノード n の興味語数である。

$$SNS_n = \frac{\sum_w tfidf_w \cdot weight_w}{NW_n} \quad (5)$$

ツリーの走査

走査は、興味ノードとその全子ノードについて行う。それらの中で最も保存ノードスコアが大きいノードを保存ノードとする。ただし、閾値 T_{SNS} を設け、最大保存ノードスコアが閾値より小さい場合は、興味ノードの子として新規に保存ノードを作成する。このようにして、ツリー構造が成長していく。

特徴語の保存

忘却処理を行った後、決定した保存ノードに、クリックしたページの特徴語とその重みを興味語として追加保存する。その後、保存ノードの全親ノードにも同じ特徴語を保存していく。同時に、補正值 M_{SN} ($0 \leq M_{SN} \leq 1$) を用いて特徴語の重みを補正していく。これは、上位ノードほど興味語の重みを小さくするためである。これにより、上位ノードほど、興味語を多く持つが各興味語の重みが小さいという親子関係が構築される。

3.4. ツリー構造の修正

使用回数が増えるほど、構造は複雑化し、忘却処理によって興味語の重みが極端に小さくなったノードや似たような興味語をもつノードが生じると考えられる。そこでツリー構造の修正を行う。

不要ノードの削除

各ノードについて保存されている興味語の重みの総和を、そのノードの重みとする。この重みが閾値 T_{DNS} 以下の場合、興味状態を表すのに相応しくないノードとみなし削除する。削除する際、削除されるノードの子ノードは親ノードの子として引継がれる。

類似度が高いノード同士の統合

ノード間の類似度を測定し統合判定を行う。類似度は、興味語の重みをノードの特徴ベクトルとしたベクトル空間法に基づき計算した。

まず、ノード n について保存されている全興味語の重みから特徴ベクトル \vec{V}_n を式(6)にもとづき求める。 w はノードに保存された興味語であり、 $weight_{\max}$ はそのノードにおける興味語の重みの最大値である。

$$\vec{V}_n = \left(\frac{weight_{w_1}}{weight_{\max}}, \frac{weight_{w_2}}{weight_{\max}}, \dots \right) \quad (6)$$

次に、式(7)から2つのノード n_1, n_2 の特徴ベクトル $\vec{V}_{n_1}, \vec{V}_{n_2}$ のなす角を求め、類似度 $sim(n_1, n_2)$ とする。

$$sim(n_1, n_2) = \frac{\vec{V}_{n_1} \cdot \vec{V}_{n_2}}{\|\vec{V}_{n_1}\| \cdot \|\vec{V}_{n_2}\|} \quad (7)$$

統合判定はまず親子間で行い、その後で兄弟間で行う。統合判定では、閾値 T_{sim} を設け、類似度が閾値以上のノードを統合する。吸収するノードは、吸収されるノードの興味語と子ノードを引継ぐ。

4. PSTree の実装

2節と3節で説明した本手法の実装について、インターフェイスとその使用法を述べる。実装には、実際の検索エンジンと同等の使用感を出すため、Perl 言語を用いてサーバ上で動く CGI プログラムとした。また、プロファイルにツリー構造を使うことから、実装した本システムを“PSTree”と名付けた。

ユーザは、まず ID とパスワードを入力し本システムにログインする必要がある。そして、検索やプロフ

PS TREE ID: iwasaki (500 times) Query 監督
 監督 Search Profile Results 5760000 hits
 2008/03/12(Tue) PM11:42 Rate: 50% / Range: 100件 / Tree Use / Not Use Time(Google) 0.620356 sec
 Time(System) 0.988005 sec

- 1 (13). [日本映画撮影監督協会](#)
 日本映画撮影監督協会オフィシャルホームページ.
 映画, 協会, 日本, 撮影, 監督,
 (44.05, 85) => 91.50 - <http://www.jpc.or.jp/> - 7k
- 2 (9). [映画監督とは - はてなダイアリー](#)
 映画監督 - ■登録-編集時に気をつけたいこと +ワードを創立させない(『←を含む+ワード』でつながるようにする)のために気をつけたいこと(はてなダイアリーヘルプ<http://db.hatena.jp/>)
 ダイアリー, ヘルプ, 創立, 気, 映画, 監督, こと,
 (32.84, 96) => 83.27 - <http://db.hatena.jp/keyword/%B1%07%B2%83%B4%0C%06%04-43k>
- ...
- 7 (1). [監督 - Wikipedia](#)
 団体競技や個人競技においては選手個々の強化や手本などを示し、コーチの役目を兼任する場合もある。一方で役割分担が明確な場合は監督業務とコーチ業務は異なる者がそれぞれを担い、一般的にはコーチは監督の指示や方針のもとに行動することが多い。 -- wikipedia, 指示, 団々, 役割, 分担, もと, こと, 行動, 手本, 者, 個人, 一般, 監督, 強化, 役目, 兼任, 業務, 団体, コーチ, 方針, 競技, 選手, (9.69, 100) => 60.99 - <http://ja.wikipedia.org/wiki/%E7%9E%A3%E7%9D%A3-22k>

図 3：インターフェイス

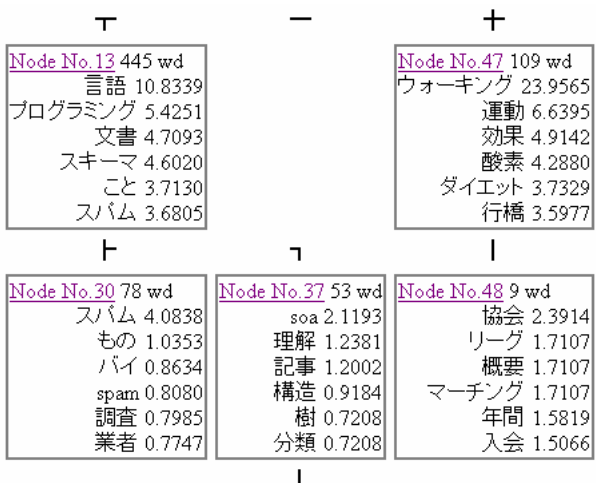


図 4：プロフィールの部分木

ファイルの閲覧などを行う。検索では個人化の比率や取得件数を指定することが可能である。図 3 に検索結果を表示したときのインターフェイスを示す。ランクの右に括弧で囲まれた数字は、元のランクである。この例では、映画に関するプロフィールを用いている。よって、同じ監督でも映画監督のページのランクが上がり、スポーツなどの監督は下がっているのがわかる。

また、作成されるプロフィールの部分木の例を図 4 に示す。表示されている語は、各ノードに保存されている興味語の上位 6 件である。ノード No.13 にはプログラミングなどコンピュータ関連の語が含まれている。その子である No.30 には、そのうちスパム関連の語が、No.37 には SOA 関連の語が保存されている。また、兄弟関係にある No.47 には異なる分野の興味語が保存されている。

5. 評価実験と考察

実装したシステムについて、再ランキングの精度評価を行った。まず、実装システムの閾値設定のための実験を行った。この実験では、パラメタを特徴的なくつかの値に設定して複数のプロフィールを作成した。そして、それぞれのプロフィールで検索し再ランキングした際の平均適合率を求めた。その後、平均適合率の高い閾値について、別のクエリを用いて同様に平均適合率を求めた。なお、検索結果の取得件数を 100 件、式(3)の個人化率 *rate* を 0.5 とし、この 100 件について再ランキングを行う。

また、プロフィールの構造が更新を重ねることで、どのように変化するかについても調べた。

5.1. プロファイルの作成

プロフィールを作成するためには、実際に検索しページをクリックする必要がある。プロフィールの作成に用いたクエリ 50 個を表 1 に示す。これは Google ディレクトリ[4]のディレクトリ名から抽出したものである。すなわち、Google ディレクトリにおいて 5 つのディレクトリ “アート”, “健康”, “コンピュータ”, “家庭”, “社会” を選び、クエリのカテゴリ名とした。そして、それぞれのサブディレクトリ名を 10 個ずつ選択し、そのカテゴリのクエリとした。検索では、タイトルとスニペットから、そのクエリのカテゴリに属すると判断できるページを 10 件クリックし、合計 500 回のプロフィールの更新を行った。このようにして、各パラメタについてプロフィールを作成していく。よって、作成したプロフィールは表 1 に示すクエリに興味をもつユーザのプロフィールと考えることができる。

表 1: プロファイルの作成に使用したクエリとそのカテゴリ

| カテゴリ | アート | 健康 | コンピュータ | 家庭 | 社会 |
|------|------------|--------|---------|-------|--------|
| クエリ | 映画 | フィットネス | インターネット | レシピ集 | ゴミ問題 |
| | 撮影所 | ヨーガ | ドメイン | シェフ | 埋め立て |
| | 演劇 | 歯科 | プログラミング | リフォーム | 平和 |
| | 歌舞伎 | 口腔外科 | C++ | 欠陥住宅 | 紛争 |
| | 配給 | ダイエット | スパム | パン | 原子力発電所 |
| | 狂言 | 歯周病 | XML | 一人暮らし | 自衛隊 |
| | フィルムコミッション | エアロビクス | プロバイダ | 弁当 | 公害 |
| | 映画祭 | ウォーキング | サーバー | 保存食 | 環境保護 |
| | 寄席 | 公衆衛生 | Perl | 引越し業者 | 生物兵器 |
| | 演芸 | 骨粗鬆症 | 自然言語処理 | 一戸建て | 核兵器 |

表 2: パラメタ設定のためのクエリと正解ページのカテゴリ

| 正解ページのカテゴリ | アート | 健康 | コンピュータ | 家庭 | 社会 |
|------------|---------|----------|----------|-----------|-------------|
| クエリ | 監督 能 | 減量 矯正 | 接続 言語 | 研究家 住居 | エネルギー 戦争 |

表 3: 正解ページ数

| 監督 | 能 | 減量 | 矯正 | 接続 | 言語 | 研究家 | 住居 | エネルギー | 戦争 | 平均 |
|----|----|----|----|----|----|-----|----|-------|----|------|
| 37 | 27 | 61 | 91 | 82 | 37 | 57 | 54 | 58 | 45 | 54.9 |

表 4: 個人化なしと一般的な個人化の平均適合率

| クエリ | 監督 | 能 | 減量 | 矯正 | 接続 | 言語 | 研究家 | 住居 | エネルギー | 戦争 | 平均 |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 個人化なし | 0.416 | 0.540 | 0.746 | 0.947 | 0.835 | 0.470 | 0.618 | 0.677 | 0.620 | 0.592 | 0.646 |
| 一般的な個人化 | 0.428 | 0.489 | 0.726 | 0.964 | 0.841 | 0.437 | 0.672 | 0.638 | 0.622 | 0.589 | 0.641 |

5.2. 平均適合率

評価指標である平均適合率は、5.1 節で作成したそれぞれのプロファイルを用いて検索し、再ランキングした検索結果から求める。また、取得した検索結果の中に全正解文書があると仮定する。パラメタ設定のための検索に用いるクエリは、同様に Google ディレクトリから選んだ。表 1 のカテゴリ名のディレクトリのサブディレクトリから、作成に用いたクエリとは別に 10 個用意し、それぞれの正解ページのカテゴリとともに表 2 に示す。

また、本手法の比較対象として、個人化を行わない場合 (Google の検索結果そのもの) と一般的な個人化を行う場合についても同様に平均適合率を求めた。一般的な個人化とは、2 節で述べた手法であり、プロファイルの構造化を行わない。また、一般的な個人化に使用するプロファイルも、5.1 節と同様に作成した。

5.3. パラメタの設定

変更するパラメタは 3.2 節と 3.3 節で述べた興味ノードスコアの閾値 T_{INS} と保存ノードスコアの閾値 T_{SNS} で、その他のパラメタは固定した。これは、この 2 つの閾値が、ページ重要度の計算やプロファイルのツリー構造の更新に大きく影響すると考えられるから

である。3.2 節と 3.3 節で述べた補正值 M_{IN} と補正值 M_{SN} は 0.5 とし、3.4 節のプロファイルの修正に用いる閾値 T_{DNS} は 1.0、閾値 T_{sim} は 0.5 とした。また、忘却係数 f は 0.99 とした。

まず、各クエリの正解ページ数を表 3 に示す。次に、個人化を行わない場合とツリー構造を持たないプロファイルによる一般的な個人化について、平均適合率をそれぞれ表 4 に示す。この結果、一般的な個人化と Google の結果を比較すると優劣はクエリごとにまちまちで、平均では一般的な個人化の方が低かった。これは、プロファイルの作成に用いた表 1 のカテゴリ間の関連性が薄いため、クエリによっては興味語がノイズとして働いたためと考えられる。

次に、構造化プロファイルを用い閾値 T_{INS} 、 T_{SNS} を変化させたときの平均適合率について述べる。図 5 は、 $T_{INS} - T_{SNS}$ 平面において平均適合率を高さとしたグラフである。グラフの青の点線は個人化なしの平均適合率を、赤の点線は一般的な個人化の平均適合率を表わす。また、グラフでの各 T_{INS} 、 T_{SNS} における平均適合率は、5 つのクエリの平均適合率の平均である。このグラフから、 T_{INS} が 3.0 以上のとき、一般的な個人化よりも平均適合率が大きくなるのがわかる。そして、一部では個人化を行わない場合よりも大きくなった。しかし、

表 5：検証実験用のクエリと正解ページのカテゴリ

| 正解ページのカテゴリ | アート | 健康 | コンピュータ | 家庭 | 社会 |
|------------|-----|--------|--------|----|------|
| クエリ | 評論 | トレーニング | アクセス | 肉類 | グリーン |

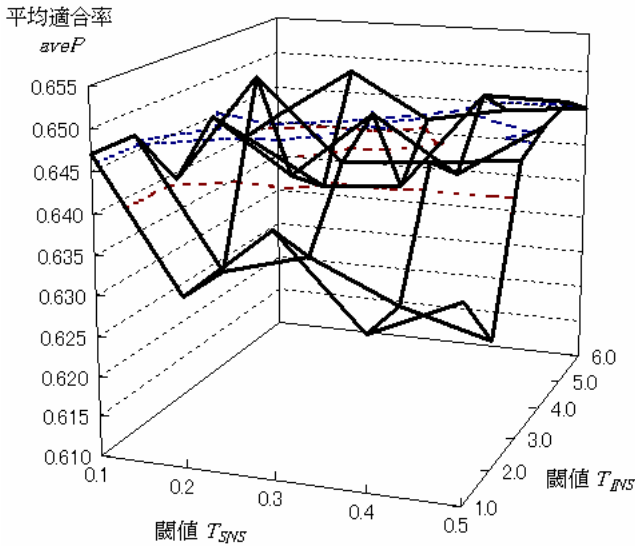


図 5： $T_I - T_S$ 平面における平均適合率

T_{INS} が 3.0 未満で T_{SNS} が 0.2 以上のときは、一様な個人化よりも平均適合率が小さくなった。よって、構造化プロフィールを用いた本システムでは、適切な閾値設定が重要であることがわかる。よって、興味語をノードごとに分類し保存することで、より正確な個人化を行える見通しを得た。

5.4. 平均適合率による評価

5.3 節で得た実験結果について、平均適合率の上位 3 点の閾値を選び、別のクエリを用いて平均適合率を求め評価を行った。用いたプロフィールは 5.1 節と同じである。クエリは、表 1 のカテゴリについてそれぞれ新しく用意した。用いたクエリを表 5 に示す。これらは、表 2 と同様に Google ディレクトリから抽出した。この実験の結果を表 6 に示す。評価に使用した閾値では、一様な個人化に比べ平均適合率が向上した。また、 $T_{INS} = 6.0, T_{SNS} = 0.2$ のときを除き、個人化を行わない場合よりも大きくなった。このことから、同じカテゴリの他のクエリでも概ね閾値が有効であることがわかる。

また、カテゴリを変更した表 1 とは別のクエリ 50 個を同様に用意し、作成したプロフィールについて、上記 3 点の閾値で平均適合率を求めた。カテゴリは“レクリエーション”，“科学”，“スポーツ”，“ゲーム”，“ビジネス”である。結果を表 7 に示す。この実験では、平均適合率は個人化なしよりも一様な個人化のほうが大きい。また本システムでは、どの閾値についても、個人化なしと一様な個人化よりも平均適合率が高

表 6：平均適合率の比較

| クエリ | 平均適合率の平均 |
|--------------------------------|----------|
| 個人化なし | 0.408 |
| 一様な個人化 | 0.404 |
| $T_{INS} = 3.0, T_{SNS} = 0.2$ | 0.409 |
| $T_{INS} = 6.0, T_{SNS} = 0.2$ | 0.405 |
| $T_{INS} = 5.0, T_{SNS} = 0.5$ | 0.412 |

表 7：別のプロフィールでの平均適合率の比較

| クエリ | 平均適合率の平均 |
|--------------------------------|----------|
| 個人化なし | 0.493 |
| 一様な個人化 | 0.565 |
| $T_{INS} = 3.0, T_{SNS} = 0.2$ | 0.571 |
| $T_{INS} = 6.0, T_{SNS} = 0.2$ | 0.569 |
| $T_{INS} = 5.0, T_{SNS} = 0.5$ | 0.567 |

くなった。よって、異なるプロフィールでも閾値が有効であることがわかる。

5.5. プロフィールのツリー構造

ユーザのプロフィールを 500 回更新するとき、どのようにプロフィールのツリー構造が変化していくかを調べた。ここで、興味ノードスコアの閾値 T_{INS} は 5.0 とし、保存ノードスコアの閾値 T_{SNS} を変化させながら、更新 50 回ごとのノード数と興味語の総数を調べた。プロフィールの生成には、表 1 のクエリを用いた。その他の条件は 5.3 節と同じである。また、閾値 T_{INS} を固定したのは、増減させてもノード数に大きな変化がみられなかったためである。

プロフィールのノード数の変化を表 8 に、興味語数の変化を表 9 に示す。興味語数は、全ノードの興味語数の和である。そのため、重みや保存されているノードが異なる同じ興味語も重複して数えている。この結果から、閾値 T_{SNS} が大きいほど新しい保存ノードが生成されやすくなり、ノード数が増えることがわかる。また、閾値 T_{SNS} が大きいほど興味語数も増加している。しかし、どちらも 350 回前後で増加率が小さくなっている。これは忘却係数によって興味語の重みが小さくなり、ノードの削除処理が行われているためと考えられる。また、5.3 節の実験において、閾値 T_{INS} が 5.0 のとき、閾値 T_{SNS} が大きいほど平均適合率は高くなった。表 8 と表 9 から、閾値 T_{SNS} が小さいほどノード数が少なく、1 つのノードに保存される興味語が多くなるこ

表 8: プロファイルのノード数

| 更新回数 | 50 | 100 | 150 | 200 | 250 | 300 | 350 | 400 | 450 | 500 |
|---------------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| T_{SNS} 0.1 | 2 | 3 | 3 | 5 | 4 | 5 | 6 | 4 | 3 | 3 |
| 0.2 | 2 | 4 | 4 | 6 | 4 | 6 | 6 | 7 | 7 | 8 |
| 0.3 | 2 | 4 | 4 | 7 | 5 | 7 | 7 | 8 | 9 | 10 |
| 0.4 | 2 | 5 | 5 | 7 | 7 | 9 | 10 | 10 | 11 | 13 |
| 0.5 | 5 | 8 | 8 | 10 | 12 | 16 | 17 | 16 | 16 | 18 |

表 9: プロファイルの興味語の総数

| 更新回数 | 50 | 100 | 150 | 200 | 250 | 300 | 350 | 400 | 450 | 500 |
|---------------|------|------|------|------|------|------|------|------|------|------|
| T_{SNS} 0.1 | 1094 | 1380 | 1888 | 2688 | 3157 | 3344 | 4140 | 4096 | 4160 | 4645 |
| 0.2 | 1094 | 1394 | 1922 | 2907 | 2714 | 3588 | 4108 | 4352 | 4741 | 5249 |
| 0.3 | 1094 | 1394 | 1922 | 2917 | 2743 | 3614 | 4134 | 4386 | 4783 | 5275 |
| 0.4 | 1094 | 1394 | 1922 | 2923 | 3494 | 4339 | 4919 | 5137 | 5548 | 6349 |
| 0.5 | 915 | 1478 | 2468 | 3166 | 3740 | 4705 | 5286 | 5527 | 5916 | 6691 |

とがわかる。よって、個人化に使用される興味語が多い場合、ノイズとして働く興味語も多くなり平均適合率が低下すると考えられる。

6. おわりに

本研究では、検索結果の再ランキングを行う個人化 Web 検索において、ユーザの多様な興味状態に対応するためプロフィールを構造化する手法を提案した。そして、その手法を実装したシステムで評価実験を行った。その結果、個人化を行わない場合、構造化を行わない一般的な個人化に比べ、個人化の精度の向上を確認した。一方、本手法では変更可能なパラメタが多く、その調整が難しい。また、プロフィールの構造については、ノード間の親子関係が不明瞭な場合もあった。よって、興味ノードスコアや保存ノードスコアの計算方法に改善の余地があると考えている。

文 献

- [1] 井上 俊, “興味傾向単語の抽出によるパーソナライズド検索システムの提案と実装,” 早稲田大学理工学部数理科学科卒業論文, February 2007.
- [2] 岩崎周造, 太田学, “興味単語を用いた個人化 Web 検索,” 情報・システムソサイエティ誌, 2007 年総合大会特別号, p.76, March 2007.
- [3] Google, “Google,” <http://www.google.com/>
- [4] Google, “Google ディレクトリ,” <http://www.google.co.jp/dirhp?hl=ja>
- [5] Yahoo!, “Yhahoo!,” <http://www.yahoo.com/>
- [6] Yhahoo!, “My Web BETA,” <http://myweb2.search.yahoo.com/>
- [7] 工藤拓, “MeCab,” <http://mecab.sourceforge.net/>