論　文

ユーザからの負担なく構築したプロファイルに基づく適応的 Web 情報検索

杉山　一成[†a)]　　波多野賢治[†]　　吉川　正俊[††]　　植村　俊亮[†]

Adaptive Web Search Based on User Profile Constructed without Any Effort from Users

Kazunari SUGIYAMA[†a)], Kenji HATANO[†], Masatoshi YOSHIKAWA[††], and Shunsuke UEMURA[†]

あらまし　Web 検索エンジンは，WWW（World Wide Web）上の情報を検索するための有用な手段である．しかし，同じ検索語が異なるユーザによって入力されたとしても，だれが検索語を入力したかにかかわらず，同じ結果を提示するという問題点を抱えている．一般に，各ユーザは自分の検索語に対して，異なる検索要求をもつと考えられる．したがって，その異なる検索要求をもつユーザに検索結果を適応させるべきであると考えられる．そこで本論文では，ユーザに負担をかけることなく各ユーザの検索要求に応じて検索結果を適応させる手法を提案し，その有効性について確かめる．実験の結果，修正した協調フィルタリングに基づいてユーザプロファイルを構築することによって，ユーザの嗜好に適応するきめの細かい検索システムを実現することができた．

キーワード　WWW，情報検索，情報フィルタリング，適合性フィードバック，ユーザモデリング

1. まえがき

インターネットの急速な普及に伴い，パーソナルコンピュータや，携帯電話，PDA（Personal Digital Assistant）などの携帯端末を用いて，だれもが容易に様々な情報を入手できるようになった．WWW 上の情報は増加し続けているため，ユーザにとって，自分の要求を満足する情報を見つけることは，ますます困難になっている．こうした状況の中で，Web 検索エンジンは，WWW 上の情報を検索するための有用な手段である．しかし，同じ検索語が異なる検索者によって入力されたとしても，だれが検索語を入力したかにかかわらず，同じ結果を提示するという問題を抱えている．一般に各ユーザは自分の検索語に対して，異なる検索要求をもつと考えられる．例えば，“Java” という検索語に対して，プログラミング言語の Java に関する文書に興味があるユーザもいれば，「コーヒー」に関する文書に興味があるユーザもいると考えられる．したがって，Web の検索結果は，異なる検索要求をもつユーザに適応させるべきであると考えられる．

こうしたシステムを実現するために，情報を個人化したり，ユーザに対してより適合する情報を提供したりする情報システムが提案されている．具体的には，(1) 適合性フィードバック [1] を用いるシステム，(2) 自分の興味や性別・年齢などの情報を登録するシステム，(3) ユーザの評価に基づいて情報を推薦するシステム，に分類される．これらのシステムにおいては，ユーザが適合か否かに関するフィードバックを行ったり，自分の興味や性別・年齢などの情報を事前に登録したり，あるいは項目に対する 1〜5 段階までの評価を行ったりする必要がある．このようなフィードバックや登録，評価を行うには時間を必要とし，負担が大きいため，ユーザはより簡単な手法を望んでいるものと考えられる．

そこで本論文では，各ユーザの検索要求に応じて検索結果を適応させるためのいくつかの手法を提案し，その検索精度を比較する．我々の手法は，ユーザに負担をかけることなく各ユーザの嗜好の変化をとらえる

† 奈良先端科学技術大学院大学情報科学研究科，生駒市
Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma-shi, 630–0192 Japan
†† 名古屋大学情報連携基盤センター，名古屋市
Information Technology Center, Nagoya University, Nagoya-shi, 464–8601 Japan
a) E-mail: kaz_sugiyama@itg.hitachi.co.jp

ことによって，よりきめの細かい検索を実現している点において新規性がある．

本論文の構成は次のとおりである．**2.** では，検索システムの個人化に着目した関連研究について述べる．**3.** では，ユーザからの負担なく各ユーザの嗜好の変化をとらえることによって，各ユーザの検索要求に適合する情報を提供するための新たな手法を提案する．**4.** では，その提案手法を評価するための実験結果を示し，その結果について考察する．最後に **5.** では，本論文のまとめと今後の課題について述べる．

2. 関 連 研 究

1. で述べたような，ユーザの検索要求に適合する情報を提供する検索システムが，これまでにも数多く提案されている．本章では，ハイパリンクに基づいた Web 検索の個人化，Web サイトの個人化，並びに推薦システムに関する研究について振り返る．

2.1 ハイパリンクに基づいた Web 検索の個人化

Web 情報検索の分野では，Web のハイパリンク構造に着目した研究が行われており，例えば，Google(注1) [2] や CLEVER プロジェクト [3] の検索エンジンを挙げることができる．また，これらの検索エンジンに関して，(1) Web ページに対する重みが単に定義されているにすぎない，(2) ハイパリンクで結ばれた Web ページ間の内容の関連性が考慮されているわけではない，といった問題点を解決し，Web ページの内容を正確に表現するために，我々はハイパリンクで結ばれた隣接ページを用いて Web ページ向けに TF-IDF 法を改良するための手法を提案した [4]〜[6]．

一方，Web のハイパリンク構造は，Web 検索の個人化においても注目されている．個人化された Web 検索を可能にする “personalized PageRank” は，Web ページの一般的な重要度を計算する PageRank アルゴリズムを修正したものとして提案されている [7]．Haveliwala [8] は，検索語の話題に適合した検索結果を提示するために，この personalized PageRank スコアを利用することによって，Web 検索の精度が改善されたと述べている．しかし，検索パターンやブックマークといった各ユーザの特徴に基づいた手法ではない．したがって，この手法による検索結果が，ユーザごとに異なる検索要求を実際に満足するかについては，明らかにされていない．

2.2 Web サイトの個人化

リンクのつながり方，及び Web ページの構造と内容は，しばしば個人化された Web サイトの構築に用いられる．本節では，リンクの個人化と内容の個人化について振り返る．

2.2.1 リンクの個人化

この手法は，ユーザの情報要求に適合するリンク先ページを選択し，ハイパリンクで結ばれた Web ページ間の関係を減らす，または改善することによって，もとの巡回空間を変化させることを意味する．電子商取引のシステムでは，顧客の購買履歴，及び評価や意見に基づいて顧客を分類することによって商品を推薦するために，この手法が用いられている．類似の商品に対して類似の評価を行うユーザは，似たような嗜好をもつと推定されるので，ユーザがある商品についての推薦を求めているとき，そのサイトは，そのユーザのクラスに対して最も人気のある商品の推薦や，そのクラスに対して与えられた商品に最もよく関係する商品の提案を行う．

インターネット上で最大規模の書店である Amazon.com(注2) の電子商取引サイトでは，ユーザが興味のありそうな新商品を伴った “New for You” ページを構築し，それを各ユーザに提示することによって，リンクの個人化が行われている．更に Amazon.com では，購入する商品を推薦するために，購買履歴を通じた暗示的な推薦や，“rate it” を通じた明示的な推薦を行う．最近の研究では，Tsandilas ら [9] は，ユーザがスライダーを操作することによって話題の重み付けを行い，その適合性に基づいて，閲覧したページにおけるリンクを自動的に個人に適応させるシステムを提案している．

2.2.2 内容の個人化

一般に，Web ページが，異なる情報を異なるユーザに提示する場合に，内容の個人化が行われる．**2.2.1** のリンクの個人化では，リンクアンカーなど Web ページの内容の一部分がユーザごとに異なる情報を提示するので，本手法とリンクの個人化との相違はとらえがたい．しかし，Web ページの内容の一部分を個人化するリンクの個人化とは異なり，ここで述べる内容の個人化は，Web ページの大部分の情報が個人化される場合を指す．例えば，My Yahoo!(注3) [10]，あるいは My Netscape(注4) は，ユーザに適合する情報をフィル

(注1)：http://www.google.com/
(注2)：http://www.amazon.com/
(注3)：http://www.my.yahoo.com/
(注4)：http://my.netscape.com/

タリングし，そのユーザが興味のありそうな項目とその詳細だけを示す．これらのサイトでは，天気，ニュース，音楽などの多数の分野から，ユーザが興味のある分野を選択し，続いてその分野の属性を選択することによって，個人化したサイトを構築することができる．更に，ユーザは自分独自のページを構築したり，個人向けにレイアウトを設定したりすることができる．しかし，ユーザの嗜好や年齢・性別といった情報は，事前のアンケートに基づいて獲得される．したがって，これらのサイトは，(1) ユーザからの入力に強く依存するため，ユーザの負担が大きくなる，(2) 事前に登録した嗜好を自分で変更しない限り，そのユーザの嗜好の変化に適応できない，といった問題点がある．

2.3 推薦システム

Web 上の有用な情報を探すことは困難になる一方であり，この状況は，しばしば「情報洪水」と呼ばれる．この情報洪水を緩和するための有望な手法の一つとして，電子商取引，電子図書館，知識管理などの分野において，推薦システムが使われ始めている．推薦システムは，ユーザの嗜好に基づいた情報を提示する．また，ユーザは項目の評価を行い，システムは推薦する項目を決定する際に，プロファイル間の類似性を利用する．推薦システムの構築においては，以下に述べる協調フィルタリングに基づいた推薦と，内容に基づいた推薦の二つの手法が主に使われている．

2.3.1 協調フィルタリングに基づいた推薦

協調フィルタリングに基づいた推薦は，これまでのところ最も成功している推薦技術である．この「協調フィルタリング」という語は，Goldberg ら [11] によって造られ，人々が読んだ文書に対する感想を記録することによって，フィルタリングを行うのに役立つように，人々が互いに協力し合うことを意味する．

この考えに基づいて，Goldberg らは最も初期に実装された協調フィルタリングに基づく推薦システム Tapestry を開発した．このシステムは電子メールのフィルタリングを行い，その際，ユーザはそのメッセージに注釈を付けることができる．しかし，Tapestry における協調フィルタリングは，自動化されておらず，ユーザは独自に設計された形式に従って，複雑な検索語を作らなければならない．また，Tapestry は，職場の社員のグループなど，結び付きの強いコミュニティにおける人々の明示的な意見を利用する．一般に，大規模なコミュニティに対する推薦システムは，互いを知るすべての人の意見を利用することは困難である．したがって，Tapestry で使われている機構は，大規模なコミュニティに対するシステムには，適切ではない．

ユーザからの評価を利用し，nearest neighbor 法などに基づく自動化された協調フィルタリングは，情報，商品，あるいはサービスなどに対して，個人向けに推薦を行い，Web 上で広く成功を収めている．GroupLens [12], [13] は，nearest neighbor 法に基づいたアルゴリズムを用いて，初めて自動化された協調フィルタリングシステムであり，Usenet ニュースのフィルタリングを行う．このシステムでは，対象とするユーザとの類似性に基づいて，そのユーザの近傍となるユーザが選択され，その対象ユーザに対してニュース記事の評価を予測し，Usenet ニュースの記事を推薦する．

Tapestry と GroupLens は明示的な評価を利用する一方，暗示的な評価を利用するシステムも存在する．例えば，Morita ら [14] は，暗示的な評価の尺度として，ニュース記事の「閲覧時間」を利用している．PHOAKS（People Helping One Another Know Stuff）[15] でも，投稿された Usenet ニュースの内容を調べることによって，Web サイトの「支持度」を暗示的な評価として利用した．その後，各ニュースグループにおいて支持度の高い Web サイトの一覧を作成する推薦システムが構築されている．

こうした暗示的な評価に基づいた推薦システムのように，ユーザからの余計な負担なくユーザの嗜好を調査するシステムもある．例えば，Letizia [16], [17] や WebWatcher [18] は，ユーザの閲覧時の振舞いを観察することによって，そのユーザの嗜好を推定する．しかし，これらのシステムでは，本来，永続的かつ緩やかに変化していくはずのユーザモデルが一定に保持されたままである．また，前述の協調フィルタリングに基づくシステムと違って，別のユーザの嗜好を考慮していない点に問題がある．

2.3.2 内容に基づいた推薦

この手法では，推薦される項目に含まれる内容を，ユーザの興味と比較することによって，推薦を行う．はじめに，ユーザの評価のモデルが確率を用いて構築され，他の項目に関するユーザの評価が与えられると，ユーザの予測の期待値を計算することによって，協調フィルタリングがどのように行われるかを推測する．ユーザの評価モデルを構築する方法として，(1) ベイジアンネットワーク [19]，(2) クラスタリング，(3) ルールに基づくモデル，の三つの異なる機械

学習手法に関して，研究が行われている．ベイジアンネットワークモデルに関しては，協調フィルタリングの課題に対して，確率モデルを構築する [20]．クラスタリングモデルは，協調フィルタリングを分類問題として扱う [20], [21]．このクラスタリングモデルでは，同じクラスの類似したユーザをクラスタリングし，特定のユーザが特定のクラスに属する確率を計算する．これをもとに，評価の条件付き確率を求める．ルールに基づくモデルは，項目間の相関を見つけるために相関ルール発見アルゴリズムを適用し，項目間の関連の強さに基づいて推薦を行う [22]．

2. 3. 1 で述べたシステムは，協調フィルタリングに基づいた推薦を行うのみであった．しかし，協調フィルタリングを内容情報と結び付けることによって，より良質な推薦を行うシステムも存在する．Fab [23] は，ユーザ間で共有される topic フィルタに加え，ユーザごとの personal フィルタを同時に構築するために，適合性フィードバックを使用する．はじめに topic フィルタによって Web ページが順位付けされ，次に personal フィルタに送られる．その後，その Web ページに対して，ユーザは適合性フィードバックを行い，このフィードバックされる情報によって，personal フィルタと topic フィルタの両方が修正される．Basu ら [21] は，推薦を分類問題として扱い，内容情報と協調フィルタリングを統合している．Melville ら [24] は，評価済みの項目の内容情報を利用することによって，協調フィルタリングの欠点を克服している．また，最近の研究では，Schafer ら [25] は，複数の情報源と推薦技術を利用して，情報量の豊かなデータを組み合わせて作られた単一の推薦リストの作成を，個人的に行うことができる新たな種類の推薦システムを提案している．

3. 提 案 手 法

2. 1 で述べたように，ハイパリンクに基づいた Web 検索の個人化は，検索結果が各ユーザの検索要求を満足するか明らかでないという問題があった．これは，閲覧パターンやブックマークといった各ユーザの特徴に基づいた個人化が行われていないことによる．**2. 2** で述べた Web サイトの個人化に関して，(1) **2. 2. 1** のリンクの個人化では，ユーザは項目を評価したり，適合する情報を得るためにスライダーを調整する必要がある，(2) **2. 2. 2** の内容の個人化では，個人の嗜好や，年齢・性別といった情報を登録する際，事前にアンケートに回答する必要があるので，ユーザの負担が大きくなる．また，自分の興味が変化すれば，登録した情報を自分で変えなければならない，といった欠点がある．更に，**2. 3** で述べた推薦システムは，ユーザが項目を積極的に評価した場合に限り，有益な推薦が行われる．しかし，項目に対するユーザの評価が，より良質な推薦を行う重要な要因であるにしても，実際にはほとんどのユーザは項目の評価を行わない．その結果，推薦の精度が不十分になり，各ユーザの検索要求に適合する情報を，必ずしも提供できるとは限らない．したがって，検索システムがより適合する情報を各ユーザに提供するためには，ユーザに負担をかけることなく，ユーザの興味の変化を直接的に，かつ正確にとらえるべきであると考えられる．このようなシステムを構築するために，我々はユーザの検索要求に応じて，検索結果を適応させる手法を提案する．**2.** で述べた研究と異なり，我々の手法は，ユーザからの負担なくユーザの嗜好の変化をとらえることによって，よりきめの細かい検索を実行できる点に新規性がある．

図 1 に，我々のシステムの概略を示す．(1) Web ブラウザを通して，ユーザが検索エンジンに検索語を入力すると，検索エンジンはその検索語に応じて，検索結果を返す．(2) その検索結果に基づいて，ユーザは自分の検索要求を満たす検索結果を選択したり，その選択した Web ページからハイパリンクをたどり，別の Web ページにアクセスして，閲覧を続けたりすることは想像にかたくない．(3) (2) におけるユーザの Web ページの閲覧過程において，我々のシステムは，ユーザの閲覧履歴を調べ，(4) 閲覧しているページが変わるたびに，そのユーザのプロファイルを更新する．(5) ユーザが次に検索語を入力する際には，ユーザプロファイルに基づいて，適合する Web ページを選択し，(6) 各ユーザに適応させた検索結果を提示する．なお，我々の実験においては，検索エンジン Google [2] の検索結果に対して，ユーザプロファイルを適用し，各

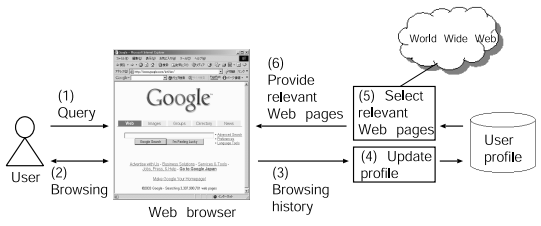

図 1　システムの概略
Fig. 1　System overview.

ユーザに適応させた検索結果を提示することとした．

以下の節では，図 1 に示すプロファイル更新（Update profile）部におけるユーザプロファイルの構築方法について説明する．我々の手法では，ユーザプロファイルは暗示的に構築される．すなわち，ユーザは自分のプロファイルを構築するために，フィードバックや評価を行うといった操作をする必要がない．我々は，以下で説明する二つの手法に基づいてユーザプロファイルを構築する．すなわち，「**3.1** 単純な閲覧履歴に基づいたユーザプロファイルの構築」により個別にプロファイルを作成し，「**3.2** 修正協調フィルタリングに基づいたユーザプロファイルの構築」によってプロファイルを改善する．

3.1 単純な閲覧履歴に基づいたユーザプロファイルの構築

本手法において，各ユーザの嗜好は，

(1) 長期間の嗜好，

(2) 1 日限りの嗜好，

の二つの側面からなるものと仮定する．長期間の嗜好においては，ユーザプロファイルは時間の経過とともに漸進的に構築され，その後の利用のために保存される．プロファイルを構築するために用いる情報は，様々な情報に基づき，ユーザの多様な側面が利用される．一方，1 日限りの嗜好においては，ユーザプロファイルを構築するために用いる情報は，そのときのセッション間のみにおいて収集され，適応処理を行うために即座に利用される．これらの二つの要素から，長期間の嗜好 $\boldsymbol{P}^{per}$ と 1 日限りの嗜好 $\boldsymbol{P}^{today}$ の両方を考慮することによって，ユーザプロファイル $\boldsymbol{P}$ を構築する．図 2 に示すように，$\boldsymbol{P}^{per}$ は，そのユーザの N 日前からの Web ページの閲覧履歴を利用して構築されるプロファイルである．ここで，$\boldsymbol{P}^{per}$ を構築するために，窓幅の概念を導入し，S_j $(j = 0, 1, 2, \cdots, N)$ を j 日目にユーザが閲覧した Web ページの数とする．図 2 に示すように，“$j = 0$” は，ユーザプロファイルを作成し始める日を意味する．各日において，$\boldsymbol{P}^{today}$ は以下のように，構築される．はじめに閲覧した Web ページ hp $(hp = 1, 2, \cdots, S_0)$ の特徴ベクトル $\boldsymbol{w}^{hp}$ を式 (1) のように表す．

$$\boldsymbol{w}^{hp} = (w^{hp}_{t_1}, w^{hp}_{t_2}, \cdots, w^{hp}_{t_m}) \tag{1}$$

ここで，m は Web ページ hp における単語の異なり数であり，t_k $(k = 1, 2, \cdots, m)$ は，各単語を表す．その各単語の頻度を用いて，$\boldsymbol{w}^{hp}$ の各要素 $w^{hp}_{t_k}$ を式 (2) のように定義する．

$$w^{hp}_{t_k} = c^{hp} \cdot \frac{tf(t_k, hp)}{\sum_{s=1}^{m} tf(t_s, hp)} \tag{2}$$

ここで，$tf(t_k, hp)$ は，閲覧した Web ページ hp における単語 t_k の頻度を表す．また，c^{hp} は各ユーザプロファイルに対して，Web ページの内容を我々のシステムが，どの程度反映するかを示す定数であり，ユーザが Web ページを閲覧する際に決定される．定数 c^{hp} は，式 (3) のように定義される．

$$c^{hp} = \begin{cases} 1; & dr \geq Th \\ 0; & dr < Th \end{cases} \tag{3}$$

ここで，dr は Web ページ hp 中の単語数によって正規化された閲読時間を表す．また，予備実験によって，しきい値 Th を 0.317 と定めた．更に，ユーザプロファイル $\boldsymbol{P}^{today}$ を，式 (4) のように表し，

$$\boldsymbol{P}^{today} = (p^{today}_{t_1}, p^{today}_{t_2}, \cdots, p^{today}_{t_m}) \tag{4}$$

各要素 $p^{today}_{t_k}$ を式 (5) のように定義する．

$$p^{today}_{t_k} = \frac{1}{S_0} \sum_{hp=1}^{S_0} w^{hp}_{t_k} \tag{5}$$

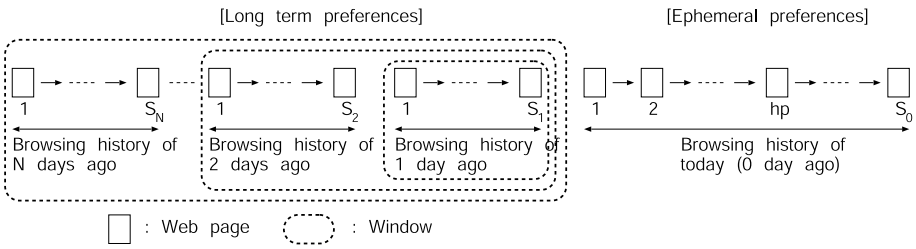

図 2 長期的なユーザプロファイルを構築するための窓幅
Fig. 2 Window size for constructing persistent user profile.

以上のようにして構築された $\boldsymbol{P}^{today}$ は，日数の経過とともに過去の閲覧履歴として蓄えられ，$\boldsymbol{P}^{per}$ の構築に用いられる．この $\boldsymbol{P}^{per}$ を構築するために，窓幅 N $(N = 1, 2, \cdots, 30)$ を設定する．$\boldsymbol{P}^{per}$ は，式 (6) のように表され，

$$\boldsymbol{P}^{per} = (p^{per}_{t_1}, p^{per}_{t_2}, \cdots, p^{per}_{t_m}) \tag{6}$$

各要素 $p^{per}_{t_k}$ を式 (7) のように定義する．

$$p^{per}_{t_k} = \frac{1}{S_N} \sum_{hp=1}^{S_N} w^{hp}_{t_k} \cdot e^{-\frac{\log 2}{hl}(d - d_{t_k\_init})} \tag{7}$$

ここで，$e^{-\frac{\log 2}{hl}(d - d_{t_k\_init})}$ は，日数の経過とともにユーザの嗜好は次第に減衰する，という仮定のもとに導入した忘却係数である．この係数において，$d_{t_k\_init}$ は，単語 t_k が最初に出現した日を，d は $d_{t_k\_init}$ に続く日数を，hl は半減期間を表す．この半減期間 hl は，7 に設定した．すなわち，ユーザの嗜好は 1 週間で半減するという考えに基づく．また，ユーザが各日に S_N ページを閲覧したと仮定する．もちろん，この閲覧した Web ページ数 S_N の値はユーザごとに異なる．したがって，式 (7) のように，S_N を用いて $p^{per}_{t_k}$ を正規化する．これらの変数を用いて，N 日の窓幅を用いた場合には，最終的に式 (8) で定義されるユーザプロファイル $\boldsymbol{P}^{(N)}$ を構築する．

$$\begin{aligned} &\boldsymbol{P}^{(N)} = a\boldsymbol{P}^{per^{(N)}} + b\boldsymbol{P}^{today} \\ &\quad \text{ただし，} \quad \boldsymbol{P}^{per^{(N)}} = \boldsymbol{P}^{(N-1)} \end{aligned} \tag{8}$$

ここで，a と b は，$a + b = 1$ を満たす定数である．なお，$\boldsymbol{P}^{today}$ が検索結果の適応に反映されるのは，$\boldsymbol{P}^{today}$ を構築した翌日である．

3.2 修正協調フィルタリングに基づいたユーザプロファイルの構築

本節では，本来の協調フィルタリングアルゴリズム，特に nearest neighbor 法に基づくアルゴリズムについて簡単に振り返り，これを修正したアルゴリズムを用いて，ユーザプロファイルを構築する方法について述べる．

3.2.1 協調フィルタリングアルゴリズムの概略

協調フィルタリングは，ユーザ–項目評価値行列において，空欄の評価値を予測する問題として表される．図 3 は，ユーザ–項目評価値行列の簡単な例を示したものである．

nearest neighbor 法に基づいたアルゴリズムは，次の各段階からなる．

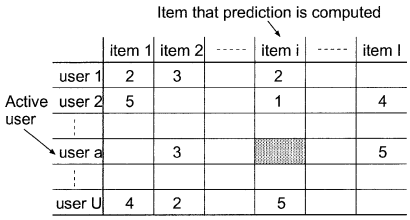

図 3 協調フィルタリングのためのユーザ–項目評価値行列
Fig. 3 User-item ratings matrix for collaborative filtering.

(i) 対象とするユーザとの類似度に関して，すべてのユーザを重み付けする．ユーザ間の類似度は，評価値ベクトル間の Pearson 相関係数として計算される．

(ii) 対象とするユーザに関して，最も高い類似度をもつ n 人のユーザを，近傍のユーザとして選択する．

(iii) 近傍のユーザの評価値の重み付けされた組合せから予測値を計算する．

段階 (i) においては，ユーザ a と u との間の類似度 $S_{a,u}$ は，式 (9) の Pearson 相関係数を用いて計算される．

$$S_{a,u} = \frac{\sum_{i=1}^{I}(r_{a,i} - \bar{r}_a) \times (r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i=1}^{I}(r_{a,i} - \bar{r}_a)^2 \times \sum_{i=1}^{I}(r_{u,i} - \bar{r}_u)^2}} \tag{9}$$

ここで，$r_{a,i}$ は，ユーザ a によって項目 i に与えられた評価値であり，$\bar{r}_a$ は，ユーザ a によって与えられた評価値の平均である．また，I は項目の総数を表す．

段階 (ii)，すなわち nearest neighbor 法に基づいたアルゴリズムでは，対象とするユーザとの類似度に基づいて，ユーザの一部が近傍のユーザとして選択される．

段階 (iii) では，評価値の重み付けされた合計が，対象とするユーザに対する項目の予測値を計算するために使用される．すなわち，式 (10) のように，近傍の平均からの重み付けされた平均偏差として，予測値が計算される．

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u=1}^{n}(r_{u,i} - \bar{r}_u) \times S_{a,u}}{\sum_{u=1}^{n} S_{a,u}} \tag{10}$$

ここで，$p_{a,i}$ は，対象とするユーザ a の項目 i に対する予測値を，$S_{a,u}$ は式 (9) で定義されるユーザ a と u の間の類似度を，n はユーザ a の近傍におけるユーザ数を表す．

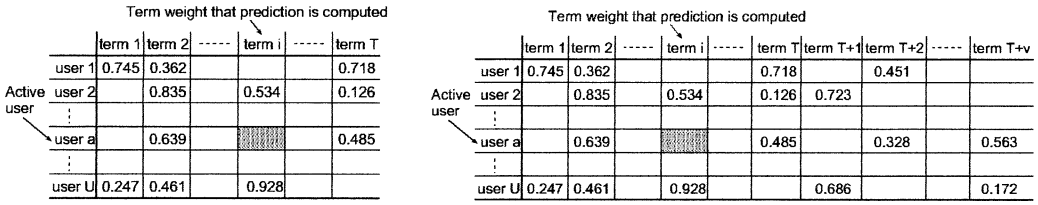

図 4　修正協調フィルタリングのためのユーザ–単語スコア行列 (a) 各ユーザが h 番目の Web ページを閲覧した場合，(b) 各ユーザが $h+1$ 番目の Web ページを閲覧した場合

Fig. 4　User-term weights matrix for modified collaborative filtering (a) when each user browsed h Web pages, (b) when each user browsed $h+1$ Web pages.

3.2.2　修正協調フィルタリングアルゴリズムを用いたユーザプロファイルの構築

3.2.1 で述べた，本来の協調フィルタリングアルゴリズムでは，ユーザ–項目評価値行列を考慮した．同様にユーザプロファイルの構築においても，図 4 (a) に示すようなユーザ–単語スコア行列を考慮することができる．また，本来の協調フィルタリングアルゴリズムに基づいて，その予測アルゴリズムを各ユーザプロファイルにおける単語のスコアを予測することに適用することができると考えられる．すなわち，ユーザプロファイルはユーザが閲覧した Web ページ中の単語のスコアに基づいて計算される．しかし，各ユーザに応じて閲覧した Web ページは異なるので，ユーザプロファイルは，図 4 に示すように，空欄のあるユーザ–単語スコア行列として構築される．すなわち，各ユーザが h ページの Web ページを閲覧することで，図 4 (a) のようなユーザ–単語スコア行列が構築される．その後，各ユーザが更に一つの Web ページを閲覧すれば，例えば図 4 (b) のように，新たに v 単語が加えられたユーザ–単語スコア行列が構築される．これは，協調フィルタリングにおけるユーザ–項目評価値行列に類似している．したがって，協調フィルタリングのアルゴリズムを用いて空欄の値を予測することによって，より正確なユーザプロファイルの構築が期待される．本手法では，我々は以下で説明する二つの手法を提案する．

(a) 固定された近傍ユーザ数に基づいたユーザプロファイルの構築

本手法において我々の提案するアルゴリズムは，次のような各段階からなる（**3.2.1** で述べた協調フィルタリングアルゴリズムとの類似性に注目されたい）．

(i)　対象とするユーザとの類似度に関して，すべてのユーザを重み付けする．ユーザ間の類似度は，**3.2.1** で述べた評価値ベクトルとは異なり，単語のスコアベクトル間の Pearson 相関係数として計算される．

(ii)　対象とするユーザに関して，最も高い類似度をもつ n 人のユーザを，近傍のユーザとして選択する．

(iii)　近傍のユーザの単語のスコアの重み付けされた組合せから予測値を計算する．

段階 (i) においては，ユーザ a と u との間の類似度 $S_{a,u}$ は，式 (11) の Pearson 相関係数を用いて計算される．

$$S_{a,u} = \frac{\sum_{i=1}^{T}(w_{a,i} - \bar{w}_a) \times (w_{u,i} - \bar{w}_u)}{\sqrt{\sum_{i=1}^{T}(w_{a,i} - \bar{w}_a)^2 \times \sum_{i=1}^{T}(w_{u,i} - \bar{w}_u)^2}} \tag{11}$$

ここで，$w_{a,i}$ は，閲覧した Web ページにおいて，式 (2) で定義される単語の頻度に基づいて計算されたユーザ a に関する i 番目の単語のスコアであり，$\bar{w}_a$ は，ユーザ a に関する単語のスコアの平均値である．また，T は単語の総数を表す．

段階 (ii)，すなわち近傍法に基づいたアルゴリズムでは，対象とするユーザとの類似度に基づいて，ユーザの一部が近傍のユーザとして選択される．この段階では，すべてのユーザに対して，選択されるユーザの数は n に固定される．したがって，我々はこの手法を「固定」と呼んでいる．

段階 (iii) では，単語のスコアの重み付けされた合計が，対象とするユーザに対する単語の予測値を計算するために使用される．すなわち，式 (12) のように，近傍の平均からの重み付けされた平均偏差として，予測値が計算される．

$$p_{a,i} = \bar{w}_a + \frac{\sum_{u=1}^{n}(w_{u,i} - \bar{w}_u) \times S_{a,u}}{\sum_{u=1}^{n} S_{a,u}} \tag{12}$$

ここで，$p_{a,i}$ は，対象とするユーザ a の単語 i の重みに対する予測値を，$S_{a,u}$ は式 (11) で定義されるユーザ a と u の間の類似度を，n はユーザ a の近傍におけるユーザ数を表す．

(b) 動的な近傍ユーザ数に基づいたユーザプロファイルの構築

本手法では，我々の提案するアルゴリズムは，次のような各段階からなる（**3. 2. 1** で述べた協調フィルタリングアルゴリズムとの類似性，及び前述の手法 (a) との相違に注目されたい）．

(i) k-nearest neighbor アルゴリズム [26]（付録参照）を用いてユーザのクラスタを作成する．手法 (a) と同様，ユーザと生成されたクラスタ間の類似度は，**3. 2. 1** で述べた評価値ベクトルとは異なり，単語のスコアベクトル間の Pearson 相関係数として計算される．

(ii) 対象とするユーザに関して，しきい値よりも高い類似度を有する n 個のクラスタを選択する．ここでは，これらの選択されたクラスタの重心ベクトルを対象とするユーザの近傍と考える．

(iii) クラスタの重心ベクトルを用いて，単語のスコアの重み付けされた組合せから予測値を計算する．

段階 (i) においては，ユーザ a と作成されたクラスタの重心ベクトル g との間の類似度 $S_{a,g}$ は，式 (13) の Pearson 相関係数を用いて計算される．

$$S_{a,g} = \frac{\sum_{i=1}^{T}(w_{a,i} - \bar{w}_a) \times (w_{g,i} - \bar{w}_g)}{\sqrt{\sum_{i=1}^{T}(w_{a,i} - \bar{w}_a)^2 \times \sum_{i=1}^{T}(w_{g,i} - \bar{w}_g)^2}} \tag{13}$$

ここで，$w_{a,i}$ は，閲覧した Web ページにおいて，式 (2) で定義される単語の頻度に基づいて計算されたユーザ a に関する i 番目の単語のスコアであり，$\bar{w}_a$ は，ユーザ a に関する単語の重みの平均値である．また，T は単語の総数を表す．

段階 (ii) では，対象としているユーザとの類似度に基づいて，一部のクラスタが選択され，その重心ベクトルを近傍のユーザとして考える．この段階では，選択されるクラスタ数はユーザごとに異なる．したがって，我々はこの手法を「動的」と呼んでいる．この手法によって各ユーザが，よりきめの細かい検索を行えるようになることが期待される．

段階 (iii) では，単語のスコアの重み付けされた合計が，対象とするユーザに対する単語の予測値を計算するために使用される．すなわち，式 (14) のように，近傍の平均からの重み付けされた平均偏差として，予測値が計算される．

$$p_{a,i} = \bar{w}_a + \frac{\sum_{g=1}^{n}(w_{g,i} - \bar{w}_g) \times S_{a,g}}{\sum_{g=1}^{n} S_{a,g}} \tag{14}$$

ここで，$p_{a,i}$ は，対象としているユーザ a の単語 i の重みに対する予測値を，$S_{a,g}$ は式 (13) で定義されるユーザ a とクラスタの重心ベクトル g の間の類似度を，n はユーザ a の近傍におけるクラスタの重心ベクトルの数を表す．

4. 評価実験

4.1 実験環境

提案手法の有効性を確かめるため，明示的にユーザプロファイルを構築する手法として，(1) 適合性フィードバック，提案手法である暗示的にユーザプロファイルを構築する手法として，(2) 簡単な閲覧履歴に基づく手法（**3. 1**），(3) 修正協調フィルタリングに基づく手法（**3. 2**），の三つの実験を行った．適合性フィードバックでは，ユーザは明示的にフィードバックをしなければならないが，我々の提案手法である (2) と (3) においては，システムが暗示的にユーザの嗜好の変化をとらえるので，ユーザには全く負担がかからない．これらの手法は，ワークステーション（CPU: UltraSparc-II 480 MHz × 4，主記憶：2 GByte, OS: Solaris8）上で Perl を用いて実装され，TREC WT10g テストコレクション [27] の検索課題として使われている 50 の検索語を用いて実験を行った．実験では，20 人の被験者の 30 日間の閲覧履歴を調査した．以下，検索結果における i 番目の Web ページを rp_i，式 (8) で定義される N 日の窓幅を用いたユーザプロファイルを $\boldsymbol{P}^{(N)}$ と表す．また，rp_i の特徴ベクトル $\boldsymbol{w}^{rp_i}$ を，式 (15) のように定義する．

$$\boldsymbol{w}^{rp_i} = (w_{t_1}^{rp_i}, w_{t_2}^{rp_i}, \cdots, w_{t_m}^{rp_i}) \tag{15}$$

ここで，m は rp_i における単語の異なり数であり，t_k は各単語を表す．その各単語の頻度を用いて $\boldsymbol{w}^{rp_i}$ の各要素 $w_{t_k}^{rp_i}$ を式 (16) のように表す．

$$w_{t_k}^{rp_i} = \frac{tf(t_k, rp_i)}{\sum_{s=1}^{m} tf(t_s, rp_i)} \tag{16}$$

ここで，$tf(t_k, rp_i)$ は rp_i における単語 t_k の頻度を表す．また，ユーザプロファイル $\boldsymbol{P}^{(N)}$ と，検索結果

における i 番目の Web ページ rp_i の特徴ベクトル $\boldsymbol{w}^{rp_i}$ 間の類似度 $sim(\boldsymbol{P}^{(N)}, \boldsymbol{w}^{rp_i})$ を式 (17) によって求める．

$$sim(\boldsymbol{P}^{(N)}, \boldsymbol{w}^{rp_i}) = \frac{\boldsymbol{P}^{(N)} \cdot \boldsymbol{w}^{rp_i}}{|\boldsymbol{P}^{(N)}| \cdot |\boldsymbol{w}^{rp_i}|} \tag{17}$$

式 (17) で得られた値によって，検索結果を各被験者のプロファイルに基づいて適応させる．この検索結果と，Google [2] の検索結果との比較を行う．なお，Google の上位 50 件の検索結果を各被験者のプロファイルに基づいて適応させた．また，検索精度の評価は R-適合率 [28] を用いて行った．検索結果の正解判定は，各被験者のプロファイルに基づいて適応させた検索結果に対して，被験者自身の判断に基づいて行われる．R の値は 30 とし，20 人の被験者の平均を計算した．

4.2 実験結果

4.2.1 適合性フィードバックに基づいたユーザプロファイル

適合性フィードバックは，検索語を修正するために最もよく使われている手法である．基本となる考えは，本来の検索語ベクトル $\boldsymbol{Q}^{org}$ が，適合文書のベクトルに近づくように，$\boldsymbol{Q}^{org}$ を新たな検索語ベクトル $\boldsymbol{Q}^{new}$ に修正することである．実験では，式 (18) で定義される Rocchio の式を用いた．

$$\boldsymbol{Q}^{new} = \alpha \boldsymbol{Q}^{org} + \frac{\beta}{|D_r|} \sum_{\boldsymbol{d}_j \in D_r} \boldsymbol{d}_j - \frac{\gamma}{|D_n|} \sum_{\boldsymbol{d}_j \in D_n} \boldsymbol{d}_j \tag{18}$$

ここで，D_r と D_n は，検索された文書の中でユーザが，それぞれ適合，非適合と判断する文書を表す．また，$|D_r|$ と $|D_n|$ は，それぞれ D_r と D_n 中の文書数を表す．なお，定数 α，β，γ の値は，それぞれ 1, 1, 1 と設定した．

我々は，検索された文書が適合か否かというユーザの判断によって得られる新たな検索語ベクトル $\boldsymbol{Q}^{new}$ が，ユーザの嗜好を反映すると考える．したがって，$\boldsymbol{Q}^{new}$ を式 (8) で定義される $\boldsymbol{P}^{today}$ として扱い，ユーザプロファイルを構築するためのユーザの初期の嗜好として用いる．この場合，式 (8) を用いて，N 日の窓幅を用いたユーザプロファイル $\boldsymbol{P}^{(N)}$ を式 (19) のように定義する．

$$\begin{aligned} &\boldsymbol{P}^{(N)} = a\boldsymbol{P}^{per^{(N)}} + b\boldsymbol{Q}^{new} \\ &\quad \text{ただし，} \quad \boldsymbol{P}^{per^{(N)}} = \boldsymbol{P}^{(N-1)} \end{aligned} \tag{19}$$

各被験者には，**4.1** で述べた 50 個の各検索語に対して，検索エンジンがそれぞれ返す検索結果が適合であるか否かを判断するように依頼し，式 (19) に基づいてユーザプロファイル $\boldsymbol{P}^{(N)}$ を構築した．ここで，$\boldsymbol{P}^{per}$ は各被験者の日常の閲覧履歴から作られたものである．

本実験において，各ユーザが行うフィードバック回数 FB を 1〜3 回まで変動させた．図 5〜図 7 は，フィードバック回数 $FB = 1, 2, 3$ の各条件下で，$a + b = 1$ を満たすように a と b の値を変動させた場合の R-適合率を示す．

4.2.2 閲覧履歴に基づいたユーザプロファイル

本手法では，N 日の窓幅を用いたユーザプロファイル $\boldsymbol{P}^{(N)}$ は **3.1** で述べたように構築され，式 (20) のように定義される．

$$\begin{aligned} &\boldsymbol{P}^{(N)} = a\boldsymbol{P}^{per^{(N)}} + b\boldsymbol{P}^{today} \\ &\quad \text{ただし，} \quad \boldsymbol{P}^{per^{(N)}} = \boldsymbol{P}^{(N-1)} \end{aligned} \tag{20}$$

図 8 は，$a + b = 1$ を満たすように a と b の値を変動させた場合の R-適合率を示す．

4.2.3 修正協調フィルタリングに基づいたユーザプロファイル

本手法においては，ユーザが新しい Web ページを閲覧する際に，新たな単語がユーザプロファイルに加えられる．しかしながら，他のユーザは必ずしも同じページを閲覧するとは限らないので，図 4 に示されるユーザ–単語スコア行列において，空欄が生じる．この空欄は，**3.2.2** で述べたアルゴリズムを用いて予測され，行列の値が満たされる．このユーザ–単語スコアベクトルはユーザの嗜好を反映すると考えられる．この予測された値を伴ったユーザ–単語スコアベクトルを $\boldsymbol{V}^{pre}$ とする．我々は $\boldsymbol{V}^{pre}$ を式 (8) で定義される $\boldsymbol{P}^{today}$ として扱い，$\boldsymbol{V}^{pre}$ をユーザプロファイルを構築するためのユーザの初期の嗜好として用いる．この場合，式 (8) を用い，N 日の窓幅を用いたユーザプロファイル $\boldsymbol{P}^{(N)}$ は式 (21) のように定義される．

$$\begin{aligned} &\boldsymbol{P}^{(N)} = a\boldsymbol{P}^{per^{(N)}} + b\boldsymbol{V}^{pre} \\ &\quad \text{ただし，} \quad \boldsymbol{P}^{per^{(N)}} = \boldsymbol{P}^{(N-1)} \end{aligned} \tag{21}$$

図 9〜図 12 は，近傍数 $n = 5, 10, 15, 20$ の各条件下で $a + b = 1$ を満たすように a と b の値を変動させた場合の R-適合率を，また，図 13 は，動的な手法の R-適合率を示す．

4.3 考察

本節では，**4.2** で述べた各手法によって得られた結

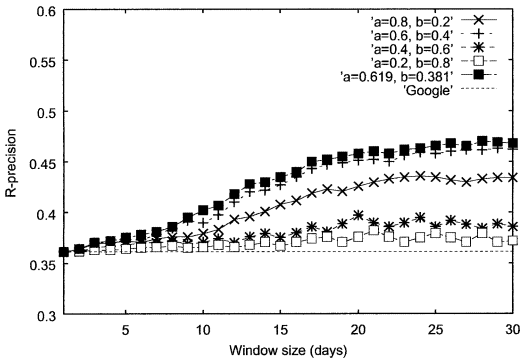

図 5 適合性フィードバックに基づいたユーザプロファイルによって得られた R-適合率 ($FB = 1$)
Fig. 5 R-precision obtained by relevance feedback-based user profile ($FB = 1$).

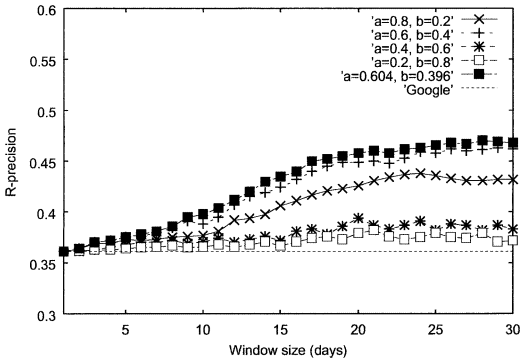

図 6 適合性フィードバックに基づいたユーザプロファイルによって得られた R-適合率 ($FB = 2$)
Fig. 6 R-precision obtained by relevance feedback-based user profile ($FB = 2$).

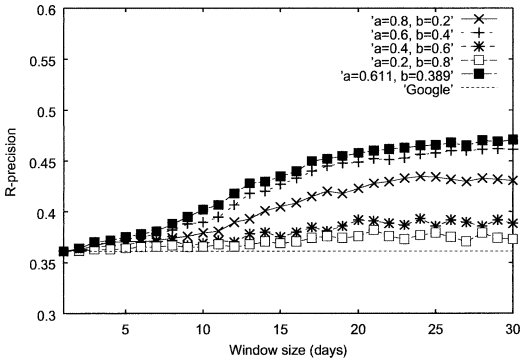

図 7 適合性フィードバックに基づいたユーザプロファイルによって得られた R-適合率 ($FB = 3$)
Fig. 7 R-precision obtained by relevance feedback-based user profile ($FB = 3$).

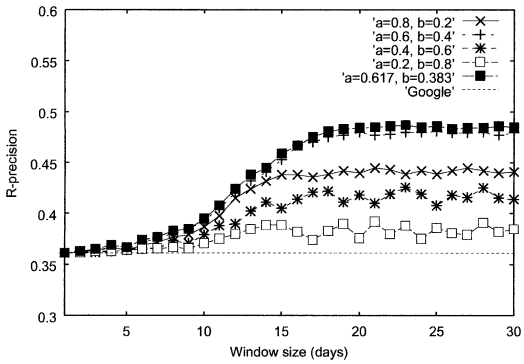

図 8 閲覧履歴に基づいたユーザプロファイルによって得られた R-適合率
Fig. 8 R-precision obtained by pure browsing history-based user profile.

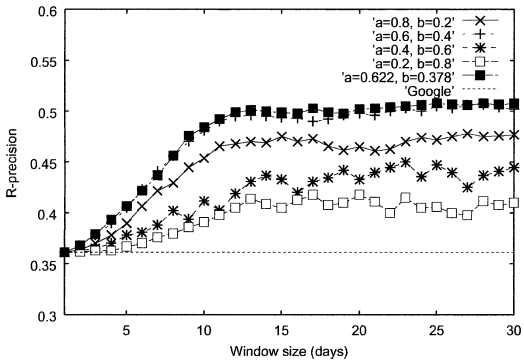

図 9 修正協調フィルタリングに基づいたユーザプロファイルによって得られた R-適合率（固定，$n = 5$）
Fig. 9 R-precision obtained by modified collaborative filtering-based user profile (static, $n = 5$).

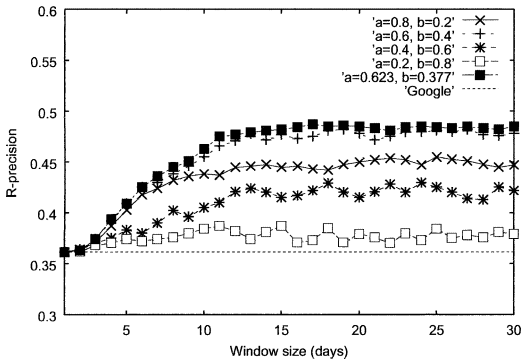

図 10 修正協調フィルタリングに基づいたユーザプロファイルによって得られた R-適合率（固定，$n = 10$）
Fig. 10 R-precision obtained by modified collaborative filtering-based user profile (static, $n = 10$).

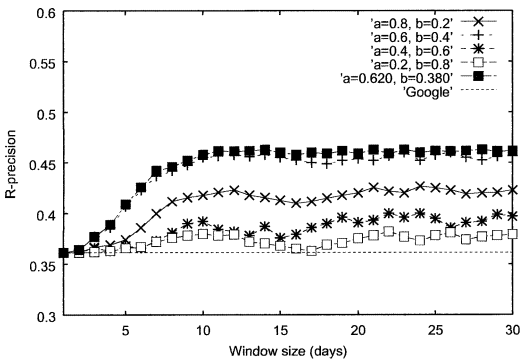

図 11　修正協調フィルタリングに基づいたユーザプロファイルによって得られた R-適合率（固定，$n = 15$）
Fig. 11　R-precision obtained by modified collaborative filtering-based user profile (static, $n = 15$).

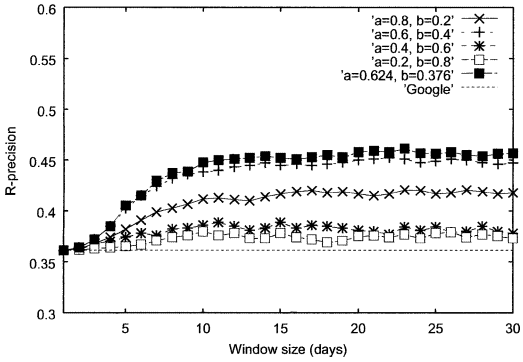

図 12　修正協調フィルタリングに基づいたユーザプロファイルによって得られた R-適合率（固定，$n = 20$）
Fig. 12　R-precision obtained by modified collaborative filtering-based user profile (static, $n = 20$).

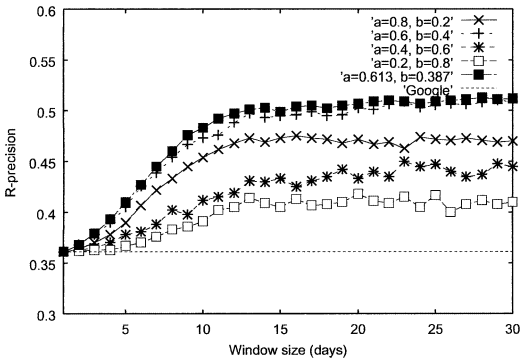

図 13　修正協調フィルタリングに基づいたユーザプロファイルによって得られた R-適合率（動的）
Fig. 13　R-precision obtained by modified collaborative filtering-based user profile (dynamic).

果について議論する．なお，図 5〜図 13 において，Google の R-適合率は図 2 に示される窓幅に依存しないため，一定となることに注意されたい．

図 5〜図 7 に示した適合性フィードバックに基づいたユーザプロファイルにおいては，フィードバック回数にかかわらず，約 20 日程度の窓幅が使われた場合に，ユーザに適応的な検索結果を返すユーザプロファイルが構築されることが分かった．**4. 2. 1** で述べたように，この手法では，初期のユーザの嗜好として，適合性フィードバックによって修正された検索語ベクトルを用いた．しかしながら，フィードバック回数を増加させても，適合率の有意な改善は得られなかった．これは，ユーザの初期の嗜好が，窓幅を用いて構築された長期間の嗜好に吸収されてしまったことによって生じた効果であると考えられる．また，1〜3 回までのフィードバック回数で，適合率は大きく改善されなかった．したがって，この範囲で実験を行ったことは，妥当であったと考えられる．

図 8 に示される単純な閲覧履歴に基づいたユーザプロファイルにおいては，約 15 日程度の窓幅が使われた場合に，ユーザに適応的な検索結果を返すユーザプロファイルが構築されることが分かった．この手法は，適合性フィードバックに基づいたユーザプロファイルによる手法よりも高い適合率が得られた．この結果は，ユーザの閲覧履歴が，そのユーザの嗜好を強く反映することを示すものであると考えられる．

更に，図 9〜図 13 に示される修正協調フィルタリングに基づいたユーザプロファイルにおいては，約 10 日程度の窓幅が使われた場合に，ユーザに適応的な検索結果を返すユーザプロファイルが構築されることが分かった．**3. 2. 2** (a) で述べた固定された近傍ユーザ数に基づいたユーザプロファイルの構築においては，図 9 に示されるように，$n = 5$，すなわち，各ユーザの近傍 5 ユーザを考慮した場合に，最適な適合率が得られた．したがって，図 10〜図 12 に示されるように，より多数の近傍のユーザを用いたとしても，検索結果を各ユーザに適応させることには，それほど寄与しないことが分かった．

3. 2. 2 で述べたように，修正協調フィルタリングに基づいた手法では，あるユーザの嗜好だけではなく，他のユーザの嗜好も利用している．このことが，固定された近傍ユーザ数に基づいたユーザプロファイルの構築において，前述の適合フィードバックや閲覧履歴に基づいた手法に比べて検索精度の向上をもたらした

ものと考えられる．また，**3. 2. 2** (b) で述べたように，動的な近傍ユーザ数に基づいたユーザプロファイルの構築においては，図 13 に示されるように，式 (21) において，$a = 0.613$，$b = 0.387$ の場合に，これまでに述べたすべての手法の実験結果の中で，最適な検索精度を得ることができた．この手法では，各ユーザの近傍ユーザは，ユーザのクラスタの重心ベクトルによって定義され，生成されるクラスタの数はユーザごとに異なる．したがって，**3. 2. 2** (a) の近傍ユーザ数を固定した手法と比較して，各ユーザはよりきめの細かい検索を行うことができるものと考えられる．

以上の考察から，明示的にフィードバックした手法が，暗示的な検索履歴よりも劣ることが判明した．適合性フィードバックを行う場合には，ユーザはある特定の検索語に対する検索結果に対してフィードバックを行う．したがって，フィードバック時には，その検索語に対してユーザがもっている一時的な印象が反映されるのみである．この事実が，フィードバックを行うことは明示的にユーザの嗜好を示すと考えられるが，暗示的な手法と比較して劣る結果になったと考えられる．

全体として，いずれの手法においても，a が 0.6 より大きく，b が 0.4 より小さい場合に最適な適合率が得られていることが分かる．これは，1 日限りの嗜好よりも長期間の嗜好をやや大きく重み付けして，ユーザプロファイルを構築した場合に，各ユーザに適応する検索結果が得られることを示すと考えられる．また，a の値がより小さく，b の値がより大きくなるにつれ，適合率の変動が大きくなることも観察される．したがって，長期間の嗜好よりも 1 日限りの嗜好を大きく重み付けして，ユーザプロファイルを構築した場合には，各ユーザに適応する検索結果を返すことは難しいと考えられる．更に，図 5〜図 13 に示されるように，我々の各提案手法は，Google によって得られた適合率を上回った．したがって，本論文で提案した手法は，これまでの検索システムでは実現されていなかった，よりきめの細かい検索が可能であると考えられる．

4. 4 ユーザプロファイルの妥当性，及び時間追従性の評価

以上の実験によって，**3.2.2** (b) で述べた修正協調フィルタリングにおける動的な近傍ユーザ数に基づいてユーザプロファイルを構築した場合に，各ユーザに対して最も適応した検索結果を提示できることが分かった．しかし，本手法の場合，他人の情報を用いているため，自分自身のユーザプロファイルによって，果たして本当に自分に適応した検索結果がもたらされるのかを評価する必要があると考えられる．そこで，表 1 に，自分自身，及び他人のユーザプロファイルを用いた場合の適合率の比較を示す．

表 1 において，各行のユーザを u_i，各列のユーザを u_j $(1 \leq i, j \leq 20)$ とすれば，表中の i 行 j 列目

表 1　他人のユーザプロファイルを用いた場合の R-適合率の比較
Table 1　Comparison of R-precision obtained using other user's profiles.

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}	u_{14}	u_{15}	u_{16}	u_{17}	u_{18}	u_{19}	u_{20}
u_1	**.504**	.375	.493	.384	.369	.381	.501	.415	.488	.482	.402	.431	.473	.428	.414	.395	.402	.423	.497	.388
u_2	.367	**.477**	.408	.403	.384	.372	.390	.394	.407	.385	.370	.412	.388	.394	.364	.413	.400	.387	.394	.396
u_3	.381	.373	**.485**	.382	.391	.405	.463	.386	.471	.469	.381	.377	.481	.391	.413	.426	.373	.378	.455	.377
u_4	.376	.402	.398	**.472**	.388	.396	.419	.431	.412	.393	.415	.385	.422	.400	.371	.382	.409	.425	.396	.405
u_5	.418	.395	.433	.450	**.518**	.411	.487	.402	.405	.431	.486	.408	.394	.465	.432	.437	.417	.392	.381	.426
u_6	.396	.411	.425	.402	.432	**.495**	.376	.398	.415	.422	.452	.389	.375	.408	.416	.377	.406	.457	.410	.393
u_7	.405	.397	.496	.416	.413	.388	**.516**	.405	.478	.503	.429	.404	.481	.423	.415	.432	.392	.373	.464	.385
u_8	.385	.379	.392	.412	.371	.426	.445	**.483**	.396	.423	.375	.412	.386	.406	.394	.446	.443	.382	.413	.407
u_9	.423	.417	.503	.376	.404	.385	.510	.399	**.515**	.482	.466	.397	.485	.420	.379	.382	.465	.441	.493	.418
u_{10}	.431	.379	.460	.399	.427	.371	.477	.373	.461	**.498**	.365	.423	.473	.416	.423	.395	.407	.423	.485	.380
u_{11}	.423	.472	.411	.422	.425	.437	.446	.432	.415	.452	**.507**	.388	.431	.386	.431	.410	.384	.467	.372	.393
u_{12}	.396	.419	.433	.387	.373	.398	.407	.418	.403	.407	.395	**.485**	.427	.461	.433	.466	.390	.421	.393	.409
u_{13}	.386	.375	.487	.394	.373	.382	.502	.391	.491	.483	.423	.377	**.519**	.428	.439	.442	.411	.412	.492	.418
u_{14}	.406	.401	.447	.435	.412	.395	.403	.415	.412	.397	.384	.405	.430	**.496**	.426	.414	.403	.386	.425	.371
u_{15}	.420	.427	.432	.398	.378	.394	.421	.396	.383	.435	.427	.381	.418	.448	**.519**	.432	.411	.426	.371	.408
u_{16}	.438	.391	.378	.389	.405	.388	.392	.460	.415	.406	.424	.423	.385	.399	.422	**.514**	.396	.374	.384	.435
u_{17}	.396	.417	.472	.418	.407	.435	.409	.412	.375	.394	.413	.399	.373	.386	.410	.381	**.503**	.421	.380	.403
u_{18}	.381	.422	.395	.423	.399	.404	.465	.425	.386	.452	.395	.406	.437	.370	.486	.423	.418	**.526**	.408	.415
u_{19}	.416	.425	.488	.452	.398	.421	.493	.409	.485	.466	.398	.403	.472	.396	.385	.412	.372	.408	**.497**	.403
u_{20}	.451	.423	.411	.376	.390	.370	.425	.428	.372	.433	.408	.426	.398	.423	.409	.433	.404	.422	.415	**.509**

の数字は，ユーザ u_i がユーザ u_j のプロファイルを用いた場合に得られた R-適合率を示す．表 1 によれば，いずれの被験者についても，自分自身のプロファイルを用いて検索結果を適応させた場合に最も良い適合率が得られており，提案手法が妥当であることが確認できた．しかし，次のような事実も観察される．例えば，u_3 が自分のプロファイルを用いた場合の適合率は 0.485 であるが，u_{13} のプロファイルを用いても 0.481 の適合率が得られている．すなわち，この結果は自分以外のプロファイルを用いても，高い適合率が得られる場合も生じることを示す．したがって，いかに本人の嗜好だけに特化させたプロファイルを構築するかが，提案手法の今後の課題として挙げられる．

また，図 5〜図 13 のグラフは，何日間の窓幅をとった場合に，最適なユーザプロファイルが作成されるかという評価であった．しかし，ユーザの嗜好は時間とともに移り変わるものである．そこで，同様に各ユーザに対して最も適応した検索結果を提示することのできる手法である **3.2.2** (b) で述べた修正協調フィルタリングにおける動的な近傍ユーザ数に基づいたユーザプロファイルの構築に関して，**4.1** で述べた被験者 20 人に対し，ユーザの嗜好の時間的な変化に追従できているかを確かめる実験を行った．各被験者が閲覧する Web ページに対して，式 (21) ($a = 0.613$, $b = 0.387$) に基づいてプロファイルを更新し，この更新されたプロファイルによって，検索結果を適応させる．なお，検索語は **4.1** と同様，TREC WT10g テストコレクション [27] の 50 の検索語を用いた．また，20 人の被験者の 30 日間のプロファイルの追従性について観察した．各日における 20 人の被験者の R-適合率についての平均を示したものが図 14 である．このグラフから，実験開始初期は，それほど高い適合率は見られない．しかし，10 日以降になると，多少の変動は観察されるものの，0.45 から 0.5 の間で増加傾向の適合率が得られており，提案手法はユーザの嗜好に追従できているものと考えられる．

4.5 他手法との比較

我々の提案手法の有効性を確かめるために，他手法との比較・検討を行う．井原ら [29] は，ユーザの潜在的好みを推定するために，好みが類似の他ユーザを探し，そのアクセス履歴を活用することにより，ユーザ本人がアクセスしていない情報集合の中に埋もれている潜在的好みに合致する情報を推薦・提示する手法を提案している．具体的には，類似ユーザを探す手法として，直接的類似ユーザ探索手法（SUSM: Similar-User Search Method）と間接的類似ユーザ探索手法（ISUSM: Indirect Similar-User Search Method）が提案されている．一方，九津見ら [30] は，インターネット上の有益な情報を簡単に入手するためのツールとして，ホームページの閲覧履歴からユーザの嗜好を学習し，その嗜好にあった新たなホームページを推薦するソフトウェアを開発している．これらの手法の詳細については，それぞれの文献を参照されたい．本節では，これらの手法と我々の提案手法との比較を行う．実験は **4.1** で述べた方法と同様の方法で行い，上述した二つの関連研究において提案されている手法と，我々の提案手法において，各ユーザに対して最も適応した検索結果を提示することのできる手法である **3.2.2** (b) で述べた修正協調フィルタリングにおける動的な近傍ユーザ数に基づいたユーザプロファイルの構築（以下，dynamic modified CF）に関して，R-適合率の比較を行った．表 2 に実験結果を示す．

いずれの手法も，各被験者のユーザプロファイルに基づいて適応させる前の Google 単独の結果と比較して，改善されている．SUSM は，他人の情報も用いるが，使用されるのは対象ユーザと最も類似度の高い

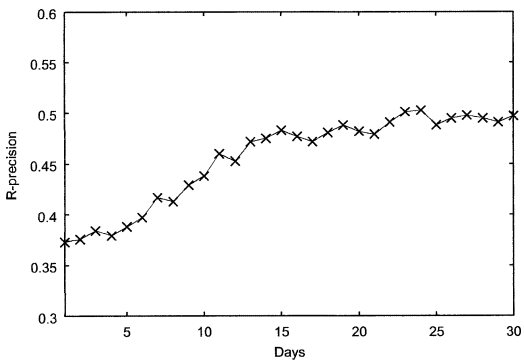

図 14　ユーザプロファイルの時間追従性
Fig. 14　Property as to whether user profile follows user's preferences that change with days.

表 2　提案手法と他手法との比較
Table 2　Comparison of our method and other methods.

	R-precision	improvement
Google	0.361	−
SUSM [29]	0.438	+0.077
ISUSM [29]	0.451	+0.09
九津見ら [30]	0.416	+0.055
dynamic modified CF	**0.513**	**+0.152**

ユーザのみである．ISUSM も SUSM と同様に他人の情報を用いるが，SUSM によって得られたユーザに基づいて，更に類似ユーザを探す際のルールがヒューリスティックに依存しているため，SUSM よりは改善されているが，その程度は顕著ではない．また，九津見らの手法を用いた場合，ユーザの特徴を学習するのに時間を必要とし，自分の嗜好情報しか使われない．これらの点が，大きな改善精度が得られなかった原因であると考えられる．一方，我々の提案手法である dynamic modified CF は，各ユーザの類似ユーザは，ユーザのクラスタの重心ベクトルによって定義され，生成されるクラスタの数も各ユーザごとに異なる．このような点が，各利用者の嗜好にきめ細かに対応でき，大きな改善精度をもたらすことができたと考えられる．したがって，以上の実験によって，我々の提案手法の優位性を確認することができたと考えられる．

5. むすび

本論文では，ユーザにより適合する情報を提供するために，ユーザの検索要求に応じて検索結果を適応させるための手法を提案した．我々の手法は，**2.** で述べた研究と比較して，ユーザに負担をかけることなく各ユーザの嗜好の変化をとらえることによって，これまでの検索システムでは実現されていなかった，きめ細かな検索を行うことができる点に新規性がある．明示的な手法として，(1) 適合性フィードバックに基づいたユーザプロファイル，暗示的な手法として，(2) 検索履歴に基づいたユーザプロファイル，(3) 修正協調フィルタリングに基づいたユーザプロファイル，を提案した．各手法によって作成されたプロファイルを用いて，検索精度を確かめるための実験，及び評価を行ったところ，修正協調フィルタリングに基づいて構築されたユーザプロファイルによって，最適な検索精度を実現することができた．この手法によって，より適切なユーザプロファイルを構築し，ユーザの嗜好にうまく適応したきめの細かい検索が実現できると考えられる．将来，ブロードバンドネットワークが幅広く普及すれば，文書だけではなく，音楽，映像など様々なメディアの情報が提供され，情報の選択肢がますます広がっていくことが容易に予想される．また，携帯電話や PDA などの携帯端末，更には ITS（Intelligent Transportation Systems）向けの車載端末に対しても，より多くの情報が提供されるようになるであろう．本論文で提案した手法は，ユーザが自分の検索要求に，より適合する情報を必要とする状況において，適用可能であると考えられる．

今後の課題として，当日の閲覧履歴を検索結果の適応に反映する手法の検討 [31]，より本人の嗜好だけに特化させたプロファイルを構築すること，更にはより多くの被験者に対する実験を行い，更に長期間のユーザの検索履歴を用いて，より各ユーザの要求に適応した検索結果を提示できるように，提案手法を改善していくことが挙げられる．

文　献

[1] J. Rocchio, “Relevance feedback in information retrieval,” in The Smart Retrieval System: Experiments in Automatic Document Processing, ed. G. Salton, pp.313–323, Prentice-Hall, Englewood Cliffs, NJ, 1971.

[2] S. Brin and L. Page, “The anatomy of a large-scale hypertextual Web search engine,” Proc. 7th International World Wide Web Conference (WWW7), pp.107–117, 1998.

[3] IBM Almaden Research Center, Clever Searching, http://www.almaden.ibm.com/cs/k53/clever.html

[4] K. Sugiyama, K. Hatano, M. Yoshikawa, and S. Uemura, “A method of improving feature vector for Web pages reflecting the contents of their out-linked pages,” Proc. 13th International Conference on Database and Expert Systems Applications (DEXA2002), pp.891–901, 2002.

[5] K. Sugiyama, K. Hatano, M. Yoshikawa, and S. Uemura, “Refinement of TF-IDF schemes for Web pages using their hyperlinked neighboring pages,” Proc. 14th ACM Conference on Hypertext and Hypermedia (HT ’03), pp.198–207, 2003.

[6] 杉山一成，波多野賢治，吉川正俊，植村俊亮，“ハイパリンクで結ばれた隣接ページの内容に基づく Web ページのための TF-IDF 法の改良，” 信学論（D-I），vol.J87-D-I, no.2, pp.113–125, Feb. 2004.

[7] L. Page, The PageRank Citation Ranking: Bringing Order to the Web, http://google.stanford.edu/~backrub/pageranksub.ps, 1998.

[8] T.H. Haveliwala, “Topic-sensitive PageRank,” Proc. 11th International World Wide Web Conference (WWW2002), pp.517–526, 2002.

[9] T. Tsandilas and M.C. Schraefel, “User-controlled link adaptation,” Proc. 14th ACM Conference on Hypertext and Hypermedia (HT ’03), pp.152–160, 2003.

[10] U. Manber, A. Patel, and J. Robison, “Experience with personalization on Yahoo!,” Commun. ACM, vol.43, no.8, pp.35–39, 2000.

[11] D. Goldberg, D. Nichols, B.M. Oki, and D.B. Terry, “Using collaborative filtering to weave an information tapestry,” Commun. ACM, vol.35, no.12, pp.61–70, 1992.

[12] P. Resnick, N. Iacovou, M. Suchak, J. Riedl, and

P. Bergstorm, “GroupLens: An open architecture for collaborative filtering of netnews,” Proc. ACM 1994 Conference on Computer Supported Cooperative Work (CSCW ’94), pp.175–186, 1994.

[13] J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon, and J. Riedl, “GroupLens: Applying collaborative filtering to usenet news,” Commun. ACM, vol.40, no.3, pp.77–87, 1997.

[14] M. Morita and Y. Shinoda, “Information filtering based on user behavior analysis and best match text retrieval,” Proc. 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR ’94), pp.272–281, 1994.

[15] L. Terveen, W. Hill, B. Amento, D. McDonald, and J. Creter, “PHOAKS: A system for sharing recommendations,” Commun. ACM, vol.40, no.3, pp.59–62, 1997.

[16] H. Lieberman, “Letizia: An agent that assists Web browsing,” Proc. 14th International Joint Conference on Artificial Intelligence (IJCAI ’95), pp.924–929, 1995.

[17] H. Lieberman, “Autonomous interface agents,” Proc. Conference on Human Factors in Computing Systems (CHI ’97), pp.67–74, 1997.

[18] T. Joachims, D. Freitag, and T.M. Mitchell, “WebWatcher: A tour guide for the World Wide Web,” Proc. 15th International Joint Conference on Artificial Intelligence (IJCAI’97), pp.770–777, 1997.

[19] J. Pearl, Probabilistic Reasoning in Intelligent Systems, Morgan Kaufmann, 1988.

[20] J.S. Breese, D. Heckerman, and C. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering,” Proc. 14th Conference on Uncertanity in Artificial Intelligence (UAI ’98), pp.43–52, 1998.

[21] C. Basu, H. Hirsh, and W. Cohen, “Recommendation as classification: Using social and content-based information in recommendation,” Proc. 15th National Conference on Artificial Intelligence (AAAI ’98), pp.714–720, 1998.

[22] B.M. Sarwar, G. Karypis, and J.A. Konstan, “Analysis of recommendation algorithms for e-commerce,” Proc. 2nd ACM Conference on Electronic Commerce (EC ’00), pp.158–167, 2000.

[23] M. Balabanovic and Y. Shoham, “Fab: Content-based, collaborative recommendation,” Commun. ACM, vol.40, no.3, pp.66–72, 1997.

[24] P. Melville, R.J. Mooney, and R. Nagarajan, “Content-boosted collaborative filtering for improved recommendations,” Proc. 18th National Conference on Artificial Intelligence (AAAI2002), pp.187–192, 2002.

[25] J.B. Schafer, J.A. Konstan, and J. Riedl, “Meta-recommendation systems: User-controlled integration of diverse recommendations,” Proc. 11th International Conference on Information and Knowledge Management (CIKM ’02), pp.43–51, 2002.

[26] R.A. Jarvis and E.A. Patrick, “Clustering using a similarity measure based on shared near neighbors,” IEEE Trans. Comput., vol.C-22, no.11, pp.1025–1034, 1973.

[27] D. Hawking, “Overview of the TREC-9 Web track,” NIST Special Publication 500-249: The Ninth Text REtrieval Conference (TREC-9), pp.87–102, 2001.

[28] R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval, ACM Press, 1999.

[29] 井原雅行, 金田洋二, 上野圭一, 金山英明, “ユーザの潜在的好み推定法,” 信学論（A）, vol.J82-A, no.5, pp.717–725, May 1999.

[30] 九津見洋, 内藤栄一, 荒木昭一, 江村里志, 新居薫治, “ユーザ適応型ホームページ推薦ソフト “ウェブナビゲーター” の開発,” 信学論（D-II）, vol.J84-D-II, no.6, pp.1149–1157, June 2001.

[31] K. Sugiyama, K. Hatano, and M. Yoshikawa, “Adaptive Web search based on user profile constructed without any effort from users,” Proc. 13th International World Wide Web Conference (WWW ’04), pp.675–684, 2004.

付　　録

k-nearest neighbor クラスタリング

$\boldsymbol{P}_i$ を各特徴ベクトルとして，N 個のパターンの集合を，$P = \{\boldsymbol{P}_1, \boldsymbol{P}_2 \cdots, \boldsymbol{P}_N\}$, 各クラスタを C_i $(i = 1, 2, \cdots)$ と表す．また，二つのパターン P_i, P_j 間の距離，パターン P_i とクラスタ C_j 間の距離を，それぞれ，$d(P_i, P_j)$，$d(P_i, C_j)$ と表す．更に，処理したパターン数と，作成したクラスタ数を，それぞれ NP, NC と表す．このとき，k-nearest neighbor クラスタリングは，以下の手順に従う．

（1） $NP = 1$, $NC = 1$ とする．また，P_{NP} を標準パターンとするクラスタ C_{NC} を作成する．

（2） 次式によって定義される距離，

$$d_j = d(\boldsymbol{P}_{(NP+1)}, C_j)$$

を求める $(1 \leq j \leq NC)$．しきい値 T よりも小さい d_j を降順にソートし，k 番目のクラスタまで，パターン $\boldsymbol{P}_{(NP+1)}$ を重複をして割り当てる．NP の値を 1 だけ増やす．

（3） $NP > N$ のとき，すべての処理を終了する．$NP \leq N$ のとき，2 に戻る．

（平成 15 年 11 月 27 日受付，16 年 3 月 30 日再受付）

杉山　一成　（正員）

1998 横浜国大･工･電子情報工学卒．2000 同大大学院電子情報工学専攻博士前期課程了．同年 KDD（現，KDDI）（株）入社，2001 退職．2004 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了．博士（工学）．現在，日立製作所ソフトウェア事業部勤務．情報検索，情報抽出に関する研究に従事．情報処理学会，人工知能学会，IEEE，ACM，AAAI 各会員．

波多野賢治　（正員）

1995 神戸大・工・計測卒．1999 同大大学院自然科学研究科博士後期課程了．博士（工学）．同大大学院自然科学研究科リサーチ・アソシエイトを経て，同年，奈良先端科学技術大学院大学情報科学研究科助手．XML データベース，Web 情報検索等の研究に従事．情報処理学会，ACM，IEEE Computer Society 各会員．

吉川　正俊　（正員）

1980 京大・工・情報工学卒．1985 同大大学院工学研究科博士後期課程了．工博．同年京都産業大学計算機科学研究所講師．同大学工学部情報通信工学科助教授，奈良先端科学技術大学院大学情報科学研究科助教授を経て，2002 名古屋大学情報連携基盤センター教授．XML データベース，多次元空間索引等の研究に従事．情報処理学会，ACM，IEEE Computer Society 各会員．

植村　俊亮　（正員）

1964 京大・工・電子卒．1966 同大大学院工学研究科修士課程了．同年通産省工業技術院電気試験所（現，産業技術総合研究所）．1988 東京農工大学工学部数理情報工学科教授．1993 奈良先端科学技術大学院大学情報科学研究科教授．工博．1970〜1971 マサチューセッツ工科大学客員研究員．データベースシステム，自然言語処理，プログラム言語の研究に従事．IEEE Fellow，情報処理学会フェロー．ACM 等各会員．