

# 発見性を考慮した協調フィルタリングアルゴリズムに関する 複数方式の検討

清水 拓也<sup>†</sup> 土方 嘉徳<sup>†</sup> 西田 正吾<sup>†</sup>

<sup>†</sup> 大阪大学大学院 基礎工学研究科システム創成専攻

〒 560-8531 大阪府豊中市待兼山 1-3

E-mail: †tshimizu@nishilab.sys.es.osaka-u.ac.jp, ††{hijikata,nishida}@sys.es.osaka-u.ac.jp

あらまし 現在, 商用 Web サイトで利用されている推薦システムのほとんどは, 協調フィルタリングを用いている. 従来の協調フィルタリングによる推薦方法では, 推薦の精度を最適化することに重点を置いているが, すでに知っているアイテムが多く推薦されるという問題があった. この推薦は, 精度を考えたときにはよい推薦であるが, ユーザの満足度を考えたときには発見性の欠如のために悪い推薦である. 本研究では, ユーザからどのようなアイテムを知っているかというデータを獲得し, このデータを用いたいくつかのアルゴリズムを提案する. これらのアルゴリズムは, ユーザの知らない好みのアイテムがどれだけ推薦されるかを示す特性である Novelty を向上させることを目的としている. 本研究では, ユーザがどのようなアイテムを知っているかという観点でのユーザ間, アイテム間の類似度を計算し, ユーザが知らないであろうアイテムを推測する. この方法と従来の嗜好に基づく協調フィルタリングを組み合わせることで, ユーザの知らない好みのアイテムを推薦しユーザの満足度を向上することを目指す.

キーワード 協調フィルタリング, 推薦システム, novelty

## Promising Methods for Discovery-oriented Collaborative Filtering

Shimizu TAKUYA<sup>†</sup>, Yoshinori HIJIKATA<sup>†</sup>, and Shogo NISHIDA<sup>†</sup>

<sup>†</sup> Graduate School of Engineering Science, Osaka University

1-3 Machikaneyama, Toyonaka, Osaka 560-8531, JAPAN

E-mail: †tshimizu@nishilab.sys.es.osaka-u.ac.jp, ††{hijikata,nishida}@sys.es.osaka-u.ac.jp

**Abstract** A number of recommender systems employed in commercial websites use collaborative filtering. The main goal of traditional techniques of collaborative filtering is improving the accuracy of the recommendation. Though, they have a problem that they include many items the user has already known. When we consider only the accuracy, these recommendations appear good. On the other hand, when we consider users' satisfactions, they are not necessarily good because of the lack of discovery. In our work, we infer items which a user does not know by computing the similarity of users or items based on what items the user has already known. We try to recommend items which the user likes and does not know combining this method and the most popular method of collaborative filtering. We hope that users' satisfactions will improve.

**Key words** recommender system, collaborative filtering, novelty

### 1. はじめに

インターネット上には大量の情報が存在している. しかし, 人々は自分の興味のあるコンテンツや商品 (以降, これらをアイテムと呼ぶ) を探すことが困難になる情報洪水の問題に直面している. 近年, この情報洪水に対処する有効な手法の一つとして推薦システムが注目されている. 推薦システムとは, ユーザの過去の嗜好, 購入履歴, デモグラフィック情報 (年齢

や性別, 地域などの人口統計的な情報) に基づいてアイテムを推薦するシステムである [1]. 推薦システムを実現する手法には, コンテンツに基づくフィルタリングと協調フィルタリングがある [2], [3]. このうち協調フィルタリングは, コンテンツを解析しなくてもすむ処理の手軽さから, 多くの方式が開発され [4] [5] [6] [7], また Amazon.com などの多くの商用サイトで用いられている [8].

伝統的に, 協調フィルタリングの研究は, 精度, 再現率や平

均絶対誤差 (MAE) を用いて、推薦の正確さを向上することに重点を置いてきた。しかし、協調フィルタリングには、その精度の高さが逆にわざわざいって、対象のユーザが既知知っているアイテムやどれも似たようなアイテムばかりが推薦されるという問題がある。そのため、ユーザはすぐに推薦に飽きてしまい、その推薦を受けなくなってしまう可能性がある。すなわち、推薦の精度が高いとしても、発見性や内容の多様性が欠落していることから、ユーザが推薦に満足しない問題が起きている [9]。我々が本稿で用いる推薦の発見性とは、ユーザの知らないアイテムがどれだけ推薦されるかを示し、推薦の多様性とは、アイテムに元々つけられているジャンルなどの分類の相違性が推薦リスト内においてどれだけ高いかを示す。例えば、あるユーザが主にモーニング娘のアルバム CD を買っていたとする。すると推薦にはモーニング娘のシングル CD や DVD、ミニモニやぷっちモニ (それぞれモーニング娘のサブグループ) の CD がリストアップされる。このとき、純粋に精度だけを考えると推薦は良いものにみえるが、新しく得る情報の少なさを考えると、ユーザは推薦に満足しないと思われる。これは特にユーザが他のアーティストや他のジャンルにも興味があった時に、より顕著となる。推薦の発見性や多様性の欠落の問題を解決することで、ユーザがより満足する推薦を行うことができると考えられる。

最近に行われた研究には、推薦に対するユーザの満足度を向上することを目的としているものがある。Ziegler ら [9] は、元々定義されているアイテムの分類を用い、推薦リスト内のトピックを多様化することでユーザの満足度を向上させている。

本研究では、ある程度の推薦精度を維持し、なおかつ推薦の発見性を向上させることによりユーザの満足度を向上させることを目指す。具体的には、発見性を考慮した新たな協調フィルタリングアルゴリズムを提案する。我々の手法と従来の協調フィルタリングの間で最も異なる点は、従来の協調フィルタリングではユーザの嗜好のプロファイルのみを利用してしたが、我々の手法ではそれに加えて、ユーザがどのようなアイテムを知っているかについてのプロファイル (以降、これを既知・不既知のプロファイルと呼ぶ) を利用することである。このプロファイルは、ユーザに事前にアイテムに既知・不既知の評価 (既知なら 1, 不既知なら 0) をつけてもらうことにより獲得する。ユーザに嗜好のみでなく既知・不既知の評価をつけてもらうためユーザの労力は大きくなるが、満足度の高い推薦が受けられるのであればある程度受け入れられると期待している。

既知・不既知のプロファイルを用いることによる本研究の貢献は以下の通りである。

- 知らないアイテムの予測 我々はユーザが知らないアイテムを予測する方法を提案する。ユーザがアイテムにつけた既知・不既知の評価から、ユーザ間、アイテム間の類似度を計算し、その類似度を用いて未評価のアイテムの既知・不既知の予測値を計算する。この予測値を基に順位付けされた知らないアイテムのリストを作成し、その予測精度を調べる。
- 嗜好と発見性を考慮した推薦 好きかつ知らないアイテムを多く推薦することを目的としたアルゴリズムをいくつか

提案する。一つ目は、嗜好の評価値と既知・不既知の評価値から統合評価値を作成し、統合評価値からなる評価値行列に協調フィルタリングを適用する。二つ目は、嗜好の予測値で順位付けされたリストと、既知・不既知の予測値で順位付けされたリストを組み合わせることを考える。三つ目は、知らないアイテムの予測を用いてユーザが知らないであろうアイテム集合を選び出し、そのアイテム集合の中から嗜好の予測評価値が高いものを推薦する。ユーザベース [5] [10] とアイテムベース [11] [12] それぞれの協調フィルタリングにおいて、これらのアルゴリズムの効果を検証する。

本稿の構成は以下の通りである。2 章では、協調フィルタリングの 2 つのアルゴリズムについて説明する。3 章では、過去に研究されてきた一般的な評価指標とユーザの満足度に関連する評価指標について説明する。4 章では、知らないアイテムの予測方法と、嗜好と発見性を考慮した協調フィルタリングアルゴリズムを示す。5 章では、本アルゴリズムの嗜好と発見性の組み合わせに対する有効性を検証する。6 章では、嗜好の評価値と既知・不既知の評価値に偏りがある場合について述べる。7 章では、本研究の関連研究を示す。8 章では、まとめと今後の課題を示す。

## 2. 協調フィルタリング

協調フィルタリングは推薦システムを作る際に最もよく使われている方法である。その基本的な考えは、ユーザと好みの似たユーザグループが好きアイテムをそのユーザに推薦するというものである。つまり、ユーザがアイテムに対して付けた評価値の統計量に基づいて、そのユーザと近い好みを持つユーザグループを特定し、それらのユーザが高く評価しているアイテムを推薦するというものである。用いられる評価値には明示的なものと暗黙的なものがある。明示的なものはユーザにアイテムについての評価を明示的に示してもらうことにより得られるものである。暗黙的なものは、購入という行動やアイテムについて言及した回数などを評価の表現として使うものである。暗黙的な評価値は一般的に収集が容易であるが、収集された情報はノイズを多く含む [13]。

### 2.1 ユーザベースの協調フィルタリング

ユーザベースの協調フィルタリングは過去 10 年間で非常に多くの研究が行われてきた。推薦の質が優れていることから、最もよく用いられる推薦アルゴリズムになっている。

ユーザ集合を  $A = \{a_1, a_2, \dots, a_n\}$ 、アイテム集合を  $B = \{b_1, b_2, \dots, b_m\}$  とし、ユーザ  $a_i$  がアイテム  $b_k$  につけた評価値を  $r_i(b_k)$  とする。

ユーザベースの協調フィルタリングの処理手順は次の 2 段階に分けることができる。

- 近傍形成  $a_i$  を注目しているユーザ (active user) としたとき、全ての  $a_o \in A \setminus \{a_i\}$  に対する類似度  $s(a_i, a_o)$  が、 $r_i$  と  $r_o$  の類似度に基づいて計算される。一般的には、 $s(a_i, a_o)$  の計算にはピアソン相関かコサイン距離が用いられる。最も似ているユーザ上位  $M$  人が  $a_i$  の近傍メンバーになり、その集合を  $neighbor(a_i) \subseteq A$  と表す。

• 評価値予測  $a_i$  の近傍メンバー  $a_o \in neighbor(a_i)$  が評価をつけており、かつ  $a_i$  が未評価であるアイテム  $b_k$  全てに対して、嗜好の予測値  $p_i(b_k)$  が計算される。予測値  $p_i(b_k)$  の値は以下のように計算される。

$$p_i(b_k) = \bar{r}_i + \frac{\sum_{a_o \in A'_i} s(a_i, a_o) * (r_o(b_k) - \bar{r}_o)}{\sum_{a_o \in A'_i} |s(a_i, a_o)|} \quad (1)$$

$$A'_i := \{a_o | a_o \in neighbor(a_i)\}$$

$$\bar{r}_i = \sum_{i=1}^n r_i(b_k) / n$$

最終的に、予測評価値  $p_i$  に基づいて上位  $N$  個の推薦リスト  $L_{p_i} : \{1, 2, \dots, N\} \rightarrow B$  が計算される。関数  $L_{p_i}$  は最も高い予測値をもつアイテムを 1 位とした降順の推薦ランキングを示す。

## 2.2 アイテムベースの協調フィルタリング

アイテムベースの協調フィルタリングは Sarwar ら [11] によって提案され、ここ 5 年ほどで活発に研究されるようになってきている。その理由には計算の複雑さの点で有利なことから、計算処理モデルを実際の予測をすることから切り離していることがあげられる。このアルゴリズムは Amazon.com が提供する推薦システムでも用いられている [8]。

アイテムベースの協調フィルタリングは、定義と予測精度の点でユーザベースに似ているが、ユーザ間ではなくアイテム間の類似度  $s$  が計算される。2 つのアイテム  $b_k, b_e$  に対して各ユーザが近い評価値を付けているとき、これらのアイテムを似ているとし類似度  $s(b_k, b_e)$  は大きい値を示す。類似度の計算にはコサイン距離を用いることが多い。各  $b_k$  に対して最も似ているアイテム上位  $M$  個が近傍  $neighbor(b_k) \subseteq B$  と定義される。予測値  $p_i(b_k)$  は以下のように計算される：

$$p_i(b_k) = \frac{\sum_{b_e \in B'_k} (s(b_k, b_e) \cdot r_i(b_e))}{\sum_{b_e \in B'_k} |s(b_k, b_e)|} \quad (2)$$

$$B'_k := \{b_e | b_e \in neighbor(b_k)\}$$

上位  $N$  個の推薦リスト  $L_{p_i}$  の最終的な計算は、ユーザベースの協調フィルタリングの手順に従う。

## 3. 評価指標

評価指標は推薦システムの質と性能を判断するために重要である。多くの評価指標は正確さの測定に関するものであり、他の要因、例えば推薦の新規性や掘り出し物を見つける性能、推薦リスト内のアイテムの多様性や発見性は考慮していない。以下の節で一般的によく使用される評価指標の概観を示す。なお、正確さの指標については [14] を参考にしている。

### 3.1 正確さの指標

正確さの指標は目的の違いから 2 つに分けられる。一つ目は、個々の予測の正確さを判断するために用いられ、アイテム  $b_k$  の予測値  $p_i(b_k)$  が、 $a_i$  の実際の評価値  $r_i(b_k)$  とどれだけ違っているかを測定する。二つ目は、予測値を基にして作成した推薦リストを評価するためのものである。一般的に 2 値の嗜好を想定している。

#### 3.1.1 予測の正確さの指標

予測の正確さの指標は、予測評価値が実際のユーザの評価値にどれだけ近いかを測定する。平均絶対誤差 (MAE) は、アイテム集合  $B_i$  の予測  $p_i(b_k)$  の正確さを統計的に測定する有効な手段であり、最も広く使われている [10] [15]。MAE は以下の式で計算される。

$$|\bar{E}| = \frac{\sum_{b_k \in B_i} |r_i(b_k) - p_i(b_k)|}{|B_i|} \quad (3)$$

MAE に関連したものに平均 2 乗誤差 (MSE) があり、これは加算の前に誤差を 2 乗する。そのため、大きい誤差は小さい誤差より目立つようになる。

予測の正確さの指標は、システム全体における予測の性能を評価することには適しているが、上位  $N$  個の推薦リストの質を評価するには不適切である。なぜなら、ユーザは高いランクのアイテムの誤差については気にするが、低いランクのアイテムの誤差には興味がないということが分かっているからである。

#### 3.1.2 決定支援指標

精度と再現率は情報検索システムの評価指標としてよく知られている。これらは予測と実際の評価値のずれを表したのではなく、むしろ推薦リストの集合が、注目しているユーザにとって必要かどうかを表したものである。

これらの指標を使う前に、ユーザ  $a_i$  によって評価付けされた全てのアイテムの集合を  $R_i$  とし、 $R_i$  をなるべく等サイズの互いに素な  $K$  個のスライスに分割する。この結果、ランダムに選ばれた  $K - 1$  個のスライスが  $a_i$  の訓練事例  $R_i^x$  を形成する。これらの評価値は、推薦を計算するために使われる  $a_i$  のプロファイルとなる。 $a_i$  の残りのスライス ( $R_i \setminus R_i^x$ ) は保持され、予測のためには使われない。このスライスはテストセットを構成する。つまり、推薦システムが予測しようとするアイテムとなる。テストセット中のアイテムの中で、ユーザが好きであるアイテムの集合を  $T_i^x$  と表す。評価値の範囲が 1-5 で与えられる場合は、評価値を 2 値に変換する必要がある。一般的には、4 または 5 である全ての評価値を”好き”、1-3 である全ての評価値を”嫌い”として 2 値化する。

Sarwar [16] は、再現率をテストセット中の好きなアイテムの総数  $|T_i^x|$  に対する、推薦リスト  $L_i^x$  中に含まれる好きなアイテム  $b \in T_i^x$  の割合として定義している：

$$Recall = 100 \cdot \frac{|T_i^x \cap \mathfrak{S}L_i^x|}{|T_i^x|} \quad (4)$$

記号  $\mathfrak{S}L_i^x$  は画像  $L_i^x$  の像であり、推薦リストの全アイテムを示す。

次に、精度は推薦リストの大きさに対する  $L_i^x$  中に含まれる好きなアイテム  $b \in T_i^x$  の割合として定義されている：

$$Precision = 100 \cdot \frac{|T_i^x \cap \mathfrak{S}L_i^x|}{|\mathfrak{S}L_i^x|} \quad (5)$$

### 3.2 正確さ以外の指標

推薦の正確さは実用性に大きな影響を与える。しかし、ユーザの満足度に影響を与える特徴には、正確さ以外のものも考えられる。正確さ以外の特徴に関する指標は、これまでの研究ではわずかしが使われていない。

### 3.2.1 Coverage

正確でない評価指標の中で、Coverage が最も頻繁に使われている [10] [18]。Coverage は、システムがどれだけアイテムを予測可能であるかを測定する指標である。システムがより多くのアイテムの予測を行えることは、そのシステムがユーザの好みのアイテムをより多く見つける可能性があることを示している。Coverage は、全アイテム数に対する予測がなされるアイテムの個数の割合として測定される。

### 3.2.2 Novelty と Serendipity

Novelty と Serendipity は推薦が当たり前でないことを示す特徴である [14]。推薦されたアイテムがユーザの知らない好みのものであるとき、この推薦は Novelty であるという。Novelty の精度と再現率をテストセット中の知らない好みのアイテムの集合  $C_i^x$  を用いて式で表すと以下ようになる：

$$\text{精度 (Novelty)} = 100 \cdot \frac{|C_i^x \cap \mathcal{S}L_i^x|}{|\mathcal{S}L_i^x|} \quad (6)$$

$$\text{再現率 (Novelty)} = 100 \cdot \frac{|C_i^x \cap \mathcal{S}L_i^x|}{|C_i^x|} \quad (7)$$

推薦が Serendipity であるということは、ユーザ自身で探すことができなかつたであろう好みのアイテムが推薦されることを意味する。Herlocker らは Serendipity の検出のためには、推薦されたアイテムがユーザをどの程度引き付け、驚きを与えたかを測定すればよいと述べている。しかし、これらを測定することは非常に難しいため、測定するためのよい指標は考えられていない。

### 3.2.3 リスト内の類似度

リスト内の類似度は推薦リストの多様性を測定する目的で考えられた指標である [9]。リスト内のアイテム間のトピックの類似度を計算し、それを合計したものをリスト内の類似度として考える。アイテム間のトピックの類似度は、ジャンルや作者、その他の特性に基づいて計算される。リスト内の類似度が高いことは、多様性が低いことを示す。

### 3.3 発見性

我々は推薦リスト内に知らないアイテムがどれだけあるかを測定するために、発見性という評価指標を提案する。発見性をリスト内のアイテム数に対するリスト内の知らないアイテム数の割合として定義する。テストセット中の知らないアイテムの集合  $D_i^x$  を用いて発見性を式で表すと以下ようになる：

$$\text{発見性} = 100 \cdot \frac{|D_i^x \cap \mathcal{S}L_i^x|}{|\mathcal{S}L_i^x|} \quad (8)$$

発見性が高いことはリスト内に知らないアイテムが多いことを示す。

## 4. 嗜好と発見性を考慮した推薦

### 4.1 知らないアイテムの予測

ユーザが知らないであろうアイテムを予測するために、1章で述べた既知・不既知のプロファイルを用いる。提案する方法は、知っているアイテムが似ている他のユーザが知っているアイテムは、注目しているユーザも知っているだろうという考

えに基づく。この考えは協調フィルタリングと同様の考えであるので、既知・不既知のプロファイルに協調フィルタリングを適用することにより、ユーザの知っているアイテムを予測できると考える。協調フィルタリングの処理手順に従い、注目するユーザと似た既知・不既知のプロファイルを持つユーザ集合を特定し、そのユーザ集合が知らないアイテムを予測するという方法を提案する。

ユーザ  $a_i$  がアイテム  $b_k$  につけた既知・不既知の評価値を  $h_i(b_k)$  とする。この評価値に対して2章で述べたユーザベース、またはアイテムベースの協調フィルタリングを適用すると、ユーザの既知・不既知の評価の予測値  $p_i^{know}(b_k)$  を計算することができる。アイテムを予測値  $p_i^{know}(b_k)$  により順位付けしたリスト  $L_{p_i^{know}}$  を作成する。 $L_{p_i^{know}}$  はユーザが知っていると予測された順でアイテムを順位付けしたリストであるので、これを逆順にソートすることで、知らないと予測された順でアイテムを順位付けしたリスト  $L_{p_i^{unknown}}$  を得ることができる。

### 4.2 嗜好と発見性を考慮した推薦アルゴリズム

本節では、いくつかの嗜好と発見性を考慮した協調フィルタリングアルゴリズムの基本的な考え方を説明し、その後でアルゴリズムの詳細を説明する。

#### 4.2.1 アルゴリズム 1

アルゴリズム 1 では、まず嗜好の評価値と既知・不既知の評価値から、好きかつ知らないアイテムが高い値を持つような統合評価値を作成することを考える。そして、この統合評価値からなる評価値行列に協調フィルタリングを適用すれば、作成される推薦リストには好みの知らないアイテムが多く含まれると考える。

統合評価値の算出には、嗜好の評価値と既知・不既知の評価値に重みをかけて足し合わせる方法をとる。まず、それぞれの評価値が統合評価値にどれだけ影響を及ぼすかを表す重みを定める。既知・不既知の評価値にける重みを  $\alpha \in [0, 1]$  とすると、嗜好の評価値にける重みを  $(1 - \alpha)$  とすることができる。このとき、嗜好の評価値と既知・不既知の評価値のスケールが等しい必要があるため、嗜好の評価値 (1~5) のスケールを 0~1 に正規化し、既知・不既知の評価値のスケールと等しくする。これらの評価値を好みの知らないアイテムが高い値を持つように足し合わせる。ユーザ  $a_i$  がアイテム  $b_k$  につけた既知・不既知の評価値を  $h_i(b_k)$ 、正規化した嗜好の評価値を  $r_i'(b_k)$  として、統合評価値  $r_i^{uni}(b_k)$  の計算式を以下のように定める。

$$r_i^{uni}(b_k) = (1 - \alpha) \times r_i'(b_k) + \alpha \times (1 - h_i(b_k)) \quad (9)$$

統合評価値  $r_i^{uni}(b_k)$  からなる評価値行列に協調フィルタリングを適用し推薦リストを作成する。先に評価値を統合しておくので、推薦リスト作成の計算コストは従来手法と同じである。

#### 4.2.2 アルゴリズム 2

嗜好のプロファイルと既知・不既知のプロファイルそれぞれに協調フィルタリングを適用して、嗜好の予測値と既知・不既知の予測値を得ることができる。この二つの予測値を組み合わせることにより、好みの知らないアイテムを多く推薦する方法を提案する。予測値の組み合わせ方にはいくつかの方法が考え

```

procedure discovery2a ( $L_{p_i}, \alpha$ ) {
   $B_i \leftarrow \mathfrak{S}L_{p_i}$ ;
   $b \in B_i$  :compute  $p_i^{know}(b)$ ;
  compute  $L_{p_i^{know}} : \{1, 2, \dots, |B_i|\} \rightarrow B_i$  using  $p_i^{know}$ ;
  for all  $b \in B_i$  do
     $L_{p_i^{unknow}}^{-1}(b) \leftarrow |B_i| - L_{p_i^{know}}^{-1}(b)$ ;
     $p_i^*(b) \leftarrow L_{p_i}^{-1}(b) \cdot (1 - \alpha) + L_{p_i^{unknow}}^{-1}(b) \cdot \alpha$ ;
  end do
  compute  $L_{p_i^*} : \{1, 2, \dots, |B_i|\} \rightarrow B_i$  using  $p_i^*$ ;
  return  $L_{p_i^*}$ ;
}

```

図 1 嗜好と発見性を考慮した協調フィルタリングアルゴリズム 2a

られ、ここでは組み合わせ方の違いで三つのアルゴリズムを提案する。

2a 嗜好の順位と不既知の順位を足し合わせる

嗜好のプロファイルに協調フィルタリングを適用して作成されるアイテムリスト  $L_{p_i}$  と、既知・不既知のプロファイルに協調フィルタリングを適用して作成されるアイテムリスト  $L_{p_i^{unknow}}$  を結合することを考える。

まず、それぞれのリストが、結果として生じるリストにどれだけ影響を及ぼすかを表す重みを  $\alpha \in [0, 1]$  とする。次に、リスト内の各アイテムについて、それぞれのリストでの順位に重みをかけたものの和を計算する。この値が小さい順にアイテムをソートすることで、リスト  $L_{p_i^*}$  を作成することができる。 $\alpha$  が大きい場合は知らないであろうアイテムを多く推薦するリストを作成し、 $\alpha$  が小さい場合は元の嗜好に基づく推薦リストに近い推薦リストを作成する。

上記の考え方に基づき考案した推薦アルゴリズムを図 1 に示す。嗜好の評価値に協調フィルタリングを適用することで得られる上位  $N$  個のリスト  $L_{p_i}$  と、先に定義した重み  $\alpha$  を入力とし、新しい推薦リスト  $L_{p_i^*}$  を出力とする。まず、リスト  $L_{p_i}$  内の全てのアイテム  $b \in B_i$  について、既知・不既知の評価の予測値  $p_i^{know}(b)$  を計算する。この予測値  $p_i^{know}(b)$  の順にアイテムを順位付けしたリスト  $L_{p_i^{know}}$  を作成する。 $L_{p_i^{know}}$  を逆順にソートすることで、知らないと予測される順にアイテムを順位付けしたリスト  $L_{p_i^{unknow}}$  を得る。ここまでの手順は 4.1 節で提案した手法である。続いて、各アイテム  $b$  に対して  $L_{p_i^{unknow}}$  での順位に重み  $\alpha$  をかけたものと  $L_{p_i}$  での順位に  $(1 - \alpha)$  をかけたものの和  $p_i^*(b)$  を計算する。 $p_i^*(b)$  の小さい順にソートすることで、最終的なリスト  $L_{p_i^*}$  を作成する。アルゴリズムでの推薦対象となるアイテムを、嗜好の予測値に基づく上位  $N$  個のリスト  $L_{p_i}$  内に含まれるアイテムに限定しているのは、明らかに嫌いと予測されるアイテムが推薦されることを防ぐためである。実際にユーザに提示する推薦リストは、上位  $N$  個のリスト  $L_{p_i^*}$  の上位  $N$  個である。ただし、入力である上位  $N$  個のリスト  $L_{w_i}$  は、ユーザに提示する上位  $N$  個のリストよりかなり大

```

procedure discovery3 ( $B_i, \alpha$ ) {
   $b \in B_i$  :compute  $p_i^{know}(b)$ ;
  set  $B'_i \leftarrow \{B_i \mid p_i^{know}(b) \leq \alpha\}$ ;
  for all  $b \in B'_i$  do
    compute  $p_i(b)$ ;
  end do
  compute  $L_{p_i} : \{1, 2, \dots, |B'_i|\} \rightarrow B_i$  using  $p_i$ ;
  return  $L_{p_i}$ ;
}

```

図 2 嗜好と発見性を考慮した協調フィルタリングアルゴリズム 3

きくなければいけないことに注意しなければいけない。

2b 嗜好の予測値と既知・不既知の予測値を足し合わせるアルゴリズム 2a では各リストでの順位を足し合わせたがり、予測値を足し合わせる方法も考えられる。各アイテムの嗜好の予測値  $p_i(b)$  と既知・不既知の予測値  $p_i^{know}(b)$  から次の式を計算する。

$$\alpha * p_i(b) + (1 - \alpha) * (1 - p_i^{know}(b)) \quad (10)$$

この値が大きいアイテムから順に並べたものを推薦リストとする。

2c 嗜好の予測値と既知・不既知の予測値の積を計算する既知・不既知の予測値  $p_i^{know}(b)$  を、ユーザ  $i$  がアイテム  $b$  を知っている確率と考えると、 $1 - p_i^{know}(b)$  はユーザ  $i$  がアイテム  $b$  を知らない確率となる。ユーザが好みで知らない確率が高いアイテムを推薦するために以下の式を計算する。

$$p_i(b) * (1 - p_i^{know}(b)) \quad (11)$$

この値が大きいアイテムから順に並べたものを推薦リストとする。嗜好の予測値が同じ値でも、知らない確率が高いアイテムが推薦されることになる。2a と 2b の方法では適切な  $\alpha$  を設定する必要があるが、この方法ではその必要がないという利点がある。

#### 4.2.3 アルゴリズム 3

アルゴリズム 2 では、嗜好の順位と不既知の順位を両方考慮している。ここで、嗜好と不既知の性質の違いについて考えてみる。嗜好に関しては、どれくらい好きかという嗜好の程度が重要である。好みのアイテムの中でも好みの程度の違いがあり、推薦システムはその中からよりユーザの好みにあったアイテムを推薦すべきである。このことから、嗜好による順位付けは必要であると言える。しかし、不既知に関しては程度は重要ではないと考える。一般的に推薦対象のアイテムの中には全く知らないアイテムが非常に多く存在しており、これらのアイテムに不既知の程度の違いはない。また、知らないアイテムの中で程度に差がある場合(例えば、全く知らない曲とどこかで聞いたことがある曲のような差)において、より知らないアイテムを推薦してほしいと望むユーザは少ないと思われる。これらの理由から、不既知に関しては順位付けすることよりも、知らないアイテム集合を特定することのほうが重要であると言える。

以上のような嗜好と不既知の性質の違いを考慮した推薦アルゴリズムを提案する．このアルゴリズムでは，知らないアイテム集合を特定し，その中に含まれるアイテムを嗜好の予測値により順位付けするという推薦方式をとる．この推薦方式に基づいて考案した推薦アルゴリズムを図 2 に示す．まず，既知・不既知のプロファイルに協調フィルタリングアルゴリズムを適用する．それにより得られる既知・不既知の予測評価値  $p_i^{know}$  ( $b$ ) が閾値  $\alpha \in [0, 1]$  より小さいアイテムの集合を知らないアイテム集合  $B'_i$  とする．次に，知らないアイテム集合  $B'_i$  に含まれるアイテムを嗜好の予測評価値  $p_i$  で順位付けすることで推薦リスト  $L_{p_i}$  を作成する．このアルゴリズムは，知らないアイテム集合をうまく特定することができれば，Novelty の向上に最も効果的であると予想される．

## 5. Novelty に関する実験

提案するアルゴリズムが Novelty を向上させるのに有効か否かを検証するために評価実験を行った．我々は次の点に注目した．

- 各提案アルゴリズムの性能

3つの提案アルゴリズムにおける，それぞれのパラメータを変化させて，Novelty と，嗜好の精度・再現率，発見性を調べる．これにより最適な  $\alpha$  の程度の値なのかを明らかにする．また，従来手法との比較も行う．

### 5.1 データセット

本実験のために，20000 の評価データを収集した．具体的には，合計 1000 曲からなる音楽データベースに対して，100 人のユーザに評価付けをしてもらった．本節では，音楽データベースの構築方法，評価データの収集方法について説明する．

筆者らは，協調フィルタリングアルゴリズムの評価を目的として，音楽データベースを構築した．本データベースには，1000 曲の音楽データを収録した．音楽データベースには，各楽曲に対する曲名，アーティスト名，リリース年，音楽ジャンル，試聴先の URL の情報を入力した．実験での音楽の使用許諾は JASRAC [19] から受け，実験での試聴データの使用許諾は，大手の商用音楽サイトから受けている．1000 曲のジャンル構成は，市場の流通量の大きさを考慮した結果，J-Pop：700 曲，演歌：75 曲，アニメ：75 曲，洋楽：150 曲とした．また J-Pop や洋楽に関しては，1960 年代から現在に至るまでの幅広い楽曲を選択した．

評価データの収集には，ユーザ 100 人（10代：13人，20代：51人，30代：5人，40代：18人，50代以上：13人）に協力してもらった．それぞれのユーザから，嗜好のプロファイルと既知・不既知のプロファイルを収集した．各ユーザごとに，音楽データベースから 200 曲をランダムに選択し，それらの曲に対して嗜好の 5 段階評価（1～5）と既知・不既知の評価（1 or 0）をつけてもらった．知らない曲に対しては試聴してもらった後に嗜好の評価をつけてもらった．一人のユーザに付き，試聴の時間を含めて，これら評価付けにかかった時間は，平均で約 3 時間である．

各ユーザが評価付けを行った 200 曲を，100 曲の訓練事例と

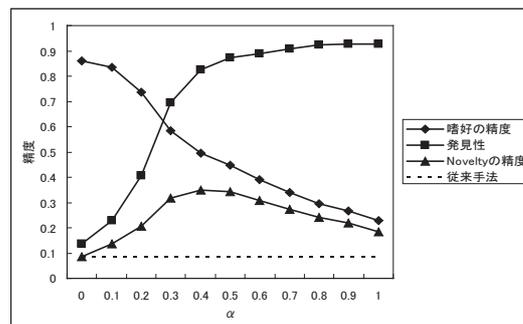


図 3 アルゴリズム 1 の各指標の推移

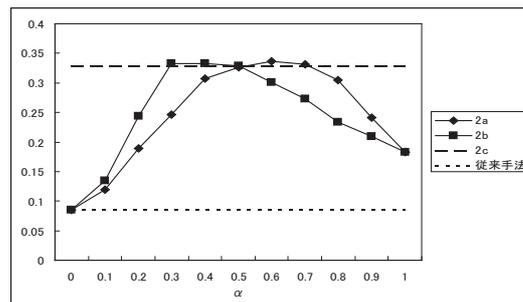


図 4 アルゴリズム 2 の各指標の推移

100 曲のテストセットに分けることで実験を行った．

### 5.2 嗜好の精度，発見性，Novelty

各提案手法において， $\alpha \in [0, 0.1, \dots, 0.9, 1.0]$  を変化させたときに作成される上位 5 個の推薦リストの各指標の変化を調べた．また，上記のように  $\alpha$  を変化させることで最適な  $\alpha$  の値を調べた．

#### 5.2.1 アルゴリズム 1

ユーザベースの協調フィルタリングに対して  $\alpha$  を変化させたときに作成される上位 5 個の推薦リストの嗜好の精度，発見性，Novelty の精度の変化を図 3 に示す． $\alpha \in [0.3, 0.4, \dots]$  のときに Novelty の精度が最も向上していることが分かる．従来手法と比較して精度は約 0.25 高い値を示している．

#### 5.2.2 アルゴリズム 2

アルゴリズム 2a, 2b, 2c と従来手法の Novelty の精度の比較を行う．ユーザベースの協調フィルタリングに対して  $\alpha$  を変化させたときに作成される上位 5 個の推薦リストの Novelty の精度の変化を図 4 に示す．アルゴリズム 2a, 2b, 2c はどれも従来手法よりも約 0.25 高い値を示していることが分かる．アルゴリズム 2a は  $\alpha = 0.3$ ，アルゴリズム 2b は  $\alpha = 0.6$  のときに Novelty が最も向上している．

#### 5.2.3 アルゴリズム 3

ユーザベースの協調フィルタリングに対して  $\alpha$  を変化させたときに作成される上位 5 個の推薦リストの嗜好の精度，発見性，Novelty の精度の変化を図 5 に示す．Novelty が最も向上しているのは， $\alpha = 0.3$  のときであり，従来手法と比較して精度は約 0.25 高い値を示している．

#### 5.2.4 各アルゴリズムの比較

どの提案アルゴリズムも従来手法と比べて Novelty の精度が

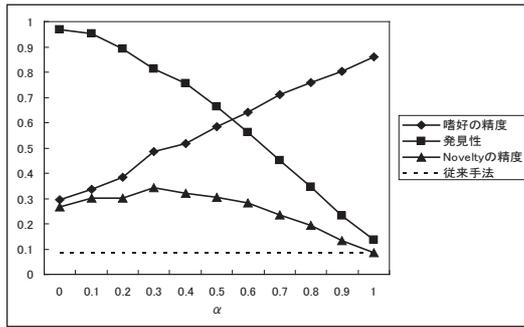


図5 アルゴリズム3の各指標の推移

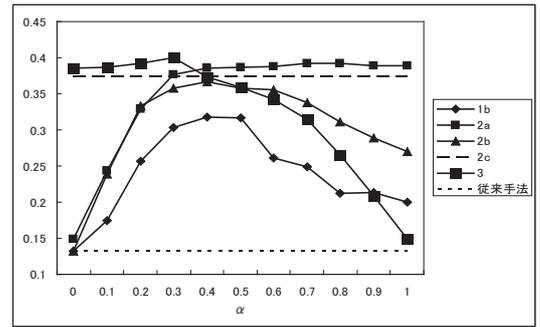


図6 評価値に偏りがある場合の Novelty の精度

向上していることが分かる。しかし、Novelty の精度の最大値はどのアルゴリズムもほぼ同じ程度であり、どのアルゴリズムが最も良いかは判断できない。

## 6. 評価値の偏り

5章の評価実験では、各アイテムに嗜好の評価値と既知・不既知の評価値の両方が必ずついている状況を想定していた。しかし、実際の利用を考えると不既知の評価値しかついていないアイテムが存在することが考えられる。ユーザが知らないアイテムに対して嗜好の評価をするときには、必ず試聴する必要がある。実際の商用サイトではユーザが全ての曲を試聴できるとは限らないこと、また試聴できる曲であっても試聴する労力がかかることから、知らないアイテムに対して必ずしも嗜好の評価がついているとは限らない。それゆえ、提案アルゴリズムはこのような評価値に偏りがある状況でも利用できる必要がある。

### 6.1 各アルゴリズムの対応

推薦対象の曲は、対象のユーザが嗜好の評価値をしていない曲とする。ユーザが嗜好の評価をしていない曲とは、対象のユーザが嗜好の評価値と既知・不既知の評価値の両方をつけていない曲と不既知の評価値のみをつけている曲になる。ユーザが不既知の評価のみをつけた曲は、ユーザはその内容を知らないため推薦対象に加えるべきであると考えられる。

#### 6.1.1 アルゴリズム1

統合評価値を計算するためには、嗜好の評価値と既知・不既知の評価値が必ず必要になる。不既知の評価値のみの曲は統合評価値が計算できないため、どちらの評価もされていない曲と同じ扱いをする。このとき、不既知の評価値のみの曲に対する既知・不既知の評価値データを全く用いないことになる。

#### 6.1.2 アルゴリズム2

推薦対象の曲は、嗜好の予測値と既知・不既知の予測値が必要となる。不既知の評価値のみの曲に対する嗜好の予測値は、どちらの評価もされていない曲と同じ方法で計算される。一方、既知・不既知の予測値に関しては、実際につけられた評価値を利用する。既知・不既知の予測値  $p_i^{k^{now}}(b) = 0$  とし、不既知の順位は1位として扱う。

#### 6.1.3 アルゴリズム3

知らないアイテム集合を選び出すことが必要である。不既知の評価値のみの曲は、実際に知らないアイテムと評価されていることが

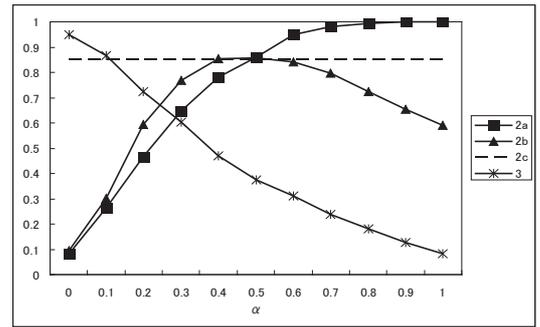


図7 不既知の評価がついていた曲の割合

ら、 $\alpha$  の値に関わらず知らないアイテム集合に加える。

## 6.2 評価実験

評価値に偏りがあるような状況での、各提案アルゴリズムの性能を評価するための実験を行った。

### 6.2.1 データセット

5章で用いた評価値データから、不既知の評価がついている曲の嗜好の評価値を部分的に消すことにより、評価値に偏りがあるデータセットを作成する。まず、各ユーザが評価付けを行った200曲を、100曲の訓練事例と100曲のテストセットに分ける。次に、訓練事例中の不既知の評価がついている曲の嗜好の評価をランダムに削除し、不既知の評価がついている曲の  $X\%$  ( $X \in [0, 25, 50, 75]$ ) に嗜好の評価値がついているデータセットを作成する。

### 6.2.2 実験結果

各提案手法により、テストセット100曲と訓練事例中の知らない評価のみの曲の中から上位5曲を推薦したときの結果を示す。 $X = 50$  のときの提案アルゴリズムの Novelty の精度を図6に示す。各アルゴリズムを比較すると、アルゴリズム1が他のアルゴリズムよりも Novelty の精度の最大値が低いことが分かる。これは、アルゴリズム1だけが不既知の評価値のみの曲のデータを用いていないことが原因だと考える。このことから、評価値に偏りがある状況ではアルゴリズム1は適切なアルゴリズムではないと考えられる。残りのアルゴリズムは Novelty の精度に大きな違いは見られない。そこで、推薦リスト中に不既知の評価値のみがついていた曲がどの程度含まれているかを調べた。 $X = 50$  のときのアルゴリズム 2a, 2b, 2c, 3 により作成された推薦リスト中で、不既知の評価がついていた曲の割合を図7に示す。アルゴリズム 2a, 2b, 2c では、Novelty の精度が最大

のときには推薦リスト中の約 80 %が不既知の評価がついていた曲であった。一方、アルゴリズム 3 は約 60 %であった。このことから、アルゴリズム 2a,2b,2c では、推薦リスト中に不既知の評価がついていた曲が多く含まれる傾向にあることが分かる。全く評価がついていない曲からも多くの推薦を受けたい場合には、アルゴリズム 3 が適していると考えられる。

## 7. 関連研究

本研究では、推薦の正確さの他に発見性を考慮している。推薦の正確さと他の評価指標との両方を考慮した研究はいくつかある。

Ziegler ら [9] は、協調フィルタリングによる推薦の多様性の欠如の問題に取り組んでいる。推薦リストを多様化する方法は、従来の協調フィルタリングにより作成されるリストと、リスト内のトピックの類似度により順位付けされたリストを組み合わせるといったものである。トピックの類似度は、作者やジャンルによる分類表を用いて計算される。

また加藤ら [20] は、潜在的に興味を持っているアイテムが推薦されにくいという、従来の推薦システムの問題点を解決するために、推薦の正確性と意外性のバランスを考慮した推薦方法を提案している。この方法では、まず商品の特徴を利用してアイテム特徴ベクトルとユーザ特徴ベクトルをカテゴリに分類している。そして、アイテム特徴ベクトルとユーザ特徴ベクトルの比較処理において、同じカテゴリ内に属するベクトルを比較する回数と、異なるカテゴリに属するベクトルを比較する回数を確率的に変化させることにより、正確性と意外性の調和のとれた推薦を実現している。

## 8. むすび

本研究では、嗜好と発見性を考慮した協調フィルタリングアルゴリズムをいくつか提案した。そして、これらの提案アルゴリズムが推薦の Novelty を向上させるのに有効であるかの評価実験を行った。実験の結果、全ての提案アルゴリズムにより Novelty の向上を確認することができた。また、不既知の評価値しかついていないアイテムが存在するときの各アルゴリズムの性能を調べた。

今後は、実際にユーザに推薦を行い、各提案アルゴリズムによる推薦に対する満足度を調査するための実験を行う。

*on Computer Supported Cooperative Work*, ACM, pp.175–186 (1994).

- [6] Shardanand, U. and Maes, P.: Social Information Filtering: Algorithms for Automating "Word of Mouth", *Proc. of CHI'95*, pp. 210–217 (1995).
- [7] Hill, W., Stead, L., Rosenstein, M. and Furnas, G.: Recommending and Evaluating Choices in a Virtual Community of Use, *Proc. of CHI'95*, pp. 194–201 (1995).
- [8] Linden, G., Smith, B., and York, J.: Amazon.com recommendations: Item-to-item collaborative filtering, *IEEE Internet Computing*, Vol.4, No.1 (2003).
- [9] Ziegler, C., McNee, S.M., Konstan, J.A. and Lausen, G.: Improving Recommendation Lists Through Topic Diversification, *WWW2005*, pp.22–32, (2005).
- [10] Herlocker, J., Konstan, J., Borchers, A. and Riedl, J.: An algorithmic framework for performing collaborative filtering, *Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM Press, pp.230–237 (1999).
- [11] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J.: Item-based collaborative filtering recommendation algorithms, *Proc. of the Tenth International World Wide Web Conference* (2001).
- [12] Deshpande, M. and Karypis, G.: Item-based top-n recommendation algorithms, *ACM Transactions on Information Systems*, Vol.22, No.1, pp.143–177 (2004).
- [13] Nichols, D.: Implicit rating and filtering, *Proc. of the Fifth DELOS Workshop on Filtering and Collaborative Filtering* (Budapest, Hungary, 1998), ERCIM, pp.31–36 (1998).
- [14] Herlocker, J., Konstan, J., Terveen, L., and Riedl, J.: Evaluating collaborative filtering recommender systems, *ACM Transactions on Information Systems*, Vol.22, No.1, pp.5–53 (2004).
- [15] Breese, J., Heckerman, D. and Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering, *Proc. of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, pp.43–52 (1998).
- [16] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J.: Application of dimensionality reduction in recommender systems, *In ACM WebKDD Workshop*(2000)
- [17] Balabanovic, M. and Shoham, Y.: Fab - content-based, collaborative recommendation, *Comm. of the ACM*, Vol.40, No.3, pp.66–72 (1997).
- [18] Middleton, S., Shadbolt, N. and De Roure, D.: Ontological user profiling in recommender systems, *ACM transactions on Information Systems*, Vol.22, No.1 pp.54–88 (2004).
- [19] 社団法人日本音楽著作権協会 JASRAC, <http://www.jasrac.or.jp/>.
- [20] 加藤由花, 川口賢二, 箱崎勝也: オンラインショッピングを対象とした正確性と意外性のバランスを考慮したリコメンダシステム, *情報処理学会論文誌*, Vol.46, No. SIG13(TOD 27), pp.53–64 (2005).

## 文 献

- [1] Resnick, P. and Varian, H.: Recommender systems, *Comm. of the ACM*, Vol.40, No.3, pp.56–58 (1997).
- [2] N. Ramakrishnan: PIPE: Web Personalization by Partial Evaluation, *IEEE Internet Computing*, Vol.4, No.6, pp.21–31 (2000).
- [3] D. Riecken.: Personalized Views of Personalization, *Comm. of the ACM*, Vol.43, No.8, pp.26–158 (2000).
- [4] Goldberg, D., Oki, B.M. and Terry, D.: Using Collaborative Filtering to Weave an information Tapestry, *Comm. of the ACM*, Vol.35, No.12, pp.61–70 (1992).
- [5] Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P. and Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews, *Proc. of the ACM 1994 Conference*