

# 一括予測型協調フィルタリング

桑田 修平<sup>†</sup> 上田 修功<sup>†</sup>

確率モデルに基づく新たな協調フィルタリング手法を提案する。提案法は、評点済みのデータから算出されるユーザごとの評点分布、アイテムごとの評点分布、および全評点に対する評点分布が、予測対象の未評点データに対して同様に算出されるそれぞれの評点分布と類似しているとの仮定に基づき、評点済みのデータに対する各評点分布と未評点データに対する各評点分布との間の KL ダイバージェンスを最小化することにより予測値を求める。評点予測を個別に独立で行う従来法と異なり、提案法は予測対象の評点に対する予測値を互いに依存させ、予測対象の評点を一括して予測できるという特長を持つ。映画に対する評価データを用いた実験の結果、予測精度は従来の代表的手法とほぼ同程度であるが、計算時間の面で顕著な優位性を確認した。

## One-shot Collaborative Filtering

SHUHEI KUWATA<sup>†</sup> and NAONORI UEDA<sup>†</sup>

We propose a new collaborative filtering method based on a probabilistic approach. In the proposed method, we assume that the empirical marginal distributions of the ratings over users and/or items are similar to these corresponding distributions of unrated data. Based on this assumption, we try to predict unobserved ratings by minimizing the Kullback-Leibler divergence between both the rated and unrated rating distributions. In contrast to the conventional methods, which predict unobserved ratings individually and independently, our method predicts all unobserved ratings simultaneously and with mutual dependence. We evaluate the prediction performance and the computational time of our method by using real movie rating data, and confirmed that the proposed method could provide prediction errors comparable to those provided by the conventional top-level methods, but could significantly reduce the computational time.

### 1. はじめに

情報フィルタリング技術の 1 つである推薦システム (Recommender system) は、大量のアイテム群の中からユーザの嗜好に合ったアイテムを自動で抽出し、そのユーザに提示するシステムである。1990 年代中頃から独立した分野としての研究が始まり、現在では理論と実務の両面から数多くの研究が行われている<sup>1),2)</sup>。実際に稼動しているシステムとして、書籍などを推薦する Amazon.com<sup>10)</sup> や、映画を推薦する GroupLens<sup>13)</sup>、楽曲を推薦する Ringo<sup>15)</sup> などがある。

推薦アイテムを抽出する技術の 1 つである協調フィルタリング (Collaborative filtering: CF) は、未購入アイテムに対するユーザの評価値 (評点) を予測す

ることでそのユーザに対する推薦アイテムを決定する技術であり、以下の 2 つのアプローチに大別される。

(1) 最近傍法に基づくアプローチ<sup>4),10),13),14),16)</sup>

推薦対象ユーザ (以降、アクティブユーザと呼ぶ) の評点履歴と類似した履歴を持つユーザ (以降、類似ユーザと呼ぶ) は、評点を予測する対象となるアイテム (以降、ターゲットアイテムと呼ぶ) に対してもアクティブユーザと類似した評点を与えるという仮定のもとで、類似ユーザのターゲットアイテムに対する既知の評点に基づいて未知の評点を予測する。ユーザ間の類似性は、ユーザの評点履歴をもとに Pearson 相関係数<sup>13)</sup> や Adjusted Cosine 類似度<sup>14)</sup> などにより計算される。

(2) 確率モデルに基づくアプローチ<sup>6),11),12),17),19)</sup>

ユーザの評点はある確率モデルに従って生成されるという仮定に基づき、全ユーザの評点履歴から統計的手法によって学習されたモデルを用いて、将来発生する評点を予測する。情報検索技術の分野で考案された pLSA (probabilistic Latent Se-

<sup>†</sup> 日本電信電話株式会社 NTT コミュニケーション科学基礎研究所  
NTT Communication Science Laboratories, NTT Corporation

mantic Analysis)<sup>7)</sup> をベースにしたモデル<sup>6),12)</sup> が代表的である。

これらの従来法は、評点予測をユーザ・アイテムごとに独立に行っている。換言すれば、ターゲットアイテムの評点を予測する際に、既知である評点履歴のみを用いている。ここで我々は、ターゲットアイテム以外の未評点アイテムに対する評点予測値も、ターゲットアイテムの評点予測に対して何らかの有用な情報を持つと考える。つまり、あるターゲットアイテムの評点を予測する際、既知である評点履歴に加え、ターゲットアイテム以外の未評点アイテムに対する予測値も考慮することで予測精度が向上すると考える。そこで本論文では、予測対象の評点に対する予測値を互いに依存させ、予測値を一括して求める一括予測型協調フィルタリング (One-shot Collaborative Filtering) 手法を提案する。

本論文の構成は以下のとおりである。まず、2章で、本論文で扱う協調フィルタリング問題を定義する。次に、3章で提案法の詳細を述べ、4章で評価実験と考察を行う。最後に5章でまとめる。

## 2. 問題設定

### 2.1 定義

$N$  人のユーザが  $M$  個のアイテムに対して与えた評点からなる  $N$  行  $M$  列の評点行列を  $R$  とする。 $R$  の  $(i, j)$  要素  $r_{i,j}$  は、 $i$  番目のユーザの  $j$  番目のアイテムに対する評点を表す。評点  $r_{i,j}$  は  $\{1, 2, \dots, V\}$  の離散値をとり、値が大きいくほど良い評価を意味するものとする。ただし、未評価である  $r_{i,j}$  の値は 0 とする。また、 $\hat{r}_{i,j}$  は  $r_{i,j}$  の予測値を表す。以降、 $i$  をユーザインデックス、 $j$  をアイテムインデックスとし、特にアクティブユーザのインデックスを  $a$ 、ターゲットアイテムのインデックスを  $t$  とする。

ユーザ  $i$  がすでに評価した (未評価の) 評点の集合を  $\mathcal{R}_{obs}^i$  ( $\mathcal{R}_{mis}^i$ ) と表す。同様に、アイテム  $j$  に対してすでに評価が与えられた (未評価の) 評点の集合を  $\mathcal{R}_{obs}^j$  ( $\mathcal{R}_{mis}^j$ ) と表す。また、評点行列  $R$  において、すでに評価が与えられた (未評価の) 評点の集合を  $\mathcal{R}_{obs}$  ( $\mathcal{R}_{mis}$ ) と表す。さらに、予測対象の評点集合を  $\mathcal{R}_{tar}$  ( $\mathcal{R}_{tar} \subseteq \mathcal{R}_{mis}$ ) とおく。

ここで、既評価の評点数  $\#\{\mathcal{R}_{obs}\}$  は、未評価の評点数  $\#\{\mathcal{R}_{mis}\}$  に比べてはるかに小さいことに注意 ( $\#\{\mathcal{R}_{obs}\} \ll \#\{\mathcal{R}_{mis}\}$ )。  $\#\{\}$  は集合の要素数を表す。CF においてベンチマークデータとしてよく用いられる映画の評価データにおいては、評点行列  $R$  の約 95% 以上の評点が未評価である。ただし、予測対象

とする評点数は既評価の評点数とほぼ同等であるものとする ( $\#\{\mathcal{R}_{tar}\} \approx \#\{\mathcal{R}_{obs}\} \ll \#\{\mathcal{R}_{mis}\}$ )。

### 2.2 目的

推薦システムの目的は、アクティブユーザが好むと期待されるアイテムを推薦することであり、通常、複数のアイテムをアクティブユーザに推薦する。これは、評点の予測値が高い順に未評点アイテムをソートし、上位にランクされた未評点アイテム群を推薦することで実現される。したがって、推薦システムの目的は、未評点アイテムに対する評点を予測する問題に帰着する。そこで、本論文における目的を、

評点行列  $R$  を用いて  $\mathcal{R}_{tar}$  の値を予測する、

と設定する。ここで、アイテムのコンテンツ情報 (アイテムが映画の場合、ジャンルや出演者名など) を用いて推薦アイテムを決定するコンテンツベースフィルタリング<sup>1)</sup> を CF と統合した手法<sup>18)</sup> や、アイテムのコンテンツ情報やユーザのデモグラフィック情報 (年齢、性別など) も含めたモデル化を行う手法<sup>5),17)</sup> など、評点以外の情報も用いた CF 手法が提案されている。しかし現状では、ユーザのデモグラフィック情報を漏れなく取得することや、異種アイテム (たとえば、映画と車) に共通するコンテンツ情報を得ることは容易ではないこと、また、デモグラフィック情報やコンテンツ情報にとらわれずに推薦アイテムを決定することが CF の特長でもあることから、本論文においては評点履歴のみからの評点予測問題を対象とする。

### 2.3 評価尺度

予測精度の評価尺度として、以下の式で表される正規化された平均絶対誤差 (Normalized Mean Absolute Error: NMAE) を用いる<sup>12)</sup>。

$$\text{NMAE} = \frac{\text{MAE}}{\text{E}[\text{MAE}]},$$

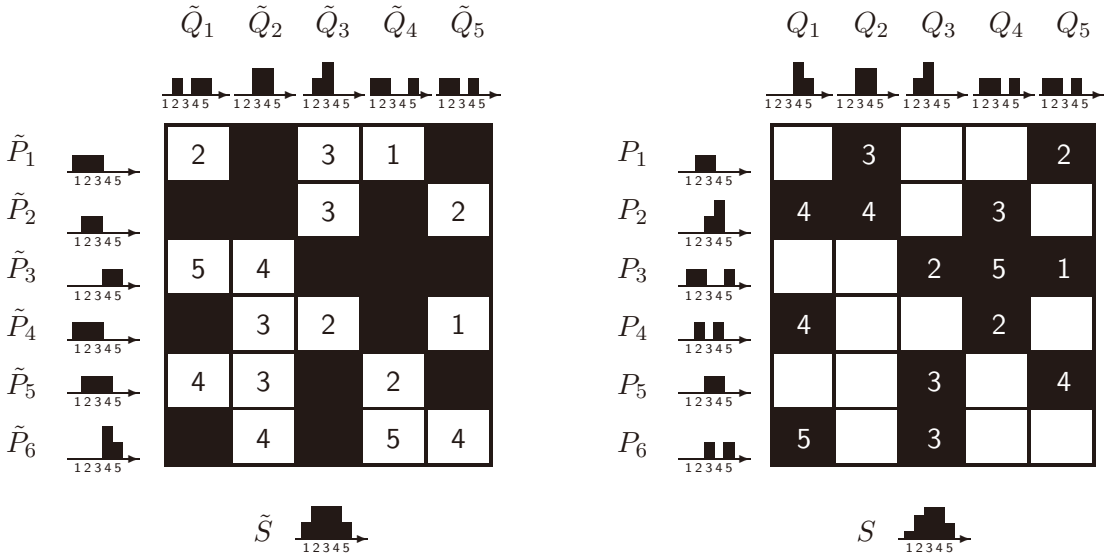
$$\text{MAE} = \frac{1}{\#\{\mathcal{R}_{tar}\}} \sum_{r_{i,j} \in \mathcal{R}_{tar}} |r_{i,j} - \hat{r}_{i,j}|.$$

ここで、 $\text{E}[\text{MAE}]$  は評点をランダムに予測したときの MAE の期待値を表し、評点のスケール  $V$  から容易に計算できる<sup>12)</sup>。NMAE の値が小さいほど良い予測精度であることを示す。NMAE が 1 より大きい (小さい) 場合、その予測手法の予測精度はランダムに予測する手法と比べて悪い (良い) と解釈できる。

## 3. 提案法

### 3.1 ランダムな欠損

文献 12) で指摘されているように、すべての CF 手法は評点行列  $R$  に対して欠損はランダムである (*Miss-*



(a) 評点済みの評点集合  $\mathcal{R}_{obs}$  に対する周辺分布

(b) 未評点の評点集合  $\mathcal{R}_{mis}$  に対する周辺評点分布

図 2 評点行列  $R$  における各周辺評点分布 ( $N = 6, M = 5, V = 5$ ). (a) 評点済みと (b) 未評点の分布間の KL ダイバージェンスを最小化することで、未評点箇所の値を予測する  
 Fig. 2 Marginal rating distributions in matrix  $R$  ( $N = 6, M = 5, V = 5$ ). The unknown ratings are predicted by minimizing the KL divergence between the (a) rated and (b) unrated rating distributions.

5 アイテム



図 1 評点行列  $R$  ( $N = 6, M = 5, V = 5$ )  
 Fig. 1 Rating Matrix  $R$  ( $N = 6, M = 5, V = 5$ ).

ing At Random: MAR) という仮定を陰においている。つまり、評点行列  $R$  において、未評点箇所はランダムに存在していると仮定している (図 1 参照)。提案法も従来法と同様に MAR の仮定をおく。すると、MAR の仮定のもとでは、評点済みの評点が従う評点分布と、未評点の評点が従う評点分布は類似していると考えられる。したがって、評点済みの評点が従う評点分布と、未評点の評点が従う評点分布との類似性を測ることにより、予測対象の評点に対する予測値の妥当性を評価することができると考えられる。

3.2 周辺評点分布に基づく評点予測

上記考えに基づき、提案法は、評点済みの評点が従う評点分布と未評点の評点が従う評点分布間の距離 (非類似性) を最小にすることで予測を行う。具体的には、まず最初に、評点行列  $R$  における評点済み評点の集合  $\mathcal{R}_{obs}$  を用いて、以下の 3 種類の評点分布、

- ユーザ周辺評点分布 (行方向で周辺化した分布)、
- アイテム周辺評点分布 (列方向で周辺化した分布)、
- ユーザ・アイテム評点分布 (行列全体の分布)、

を求める (図 2 (a) 参照)。ここで、評点済みユーザー周辺評点分布を  $\tilde{P}_i$  ( $i = 1, 2, \dots, N$ )、評点済みアイテム周辺評点分布を  $\tilde{Q}_j$  ( $j = 1, 2, \dots, M$ )、評点済みユーザー・アイテム評点分布を  $\tilde{S}$  とする。それぞれの評点分布は以下の式により計算する。

$$\tilde{P}_i(v) = \frac{\sum_{r_{i,j} \in \mathcal{R}_{obs}^i} \delta(r_{i,j} = v) + \eta}{\sum_l \sum_{r_{i,j} \in \mathcal{R}_{obs}^i} \delta(r_{i,j} = l) + \eta V}, \quad (1)$$

$$\tilde{Q}_j(v) = \frac{\sum_{r_{i,j} \in \mathcal{R}_{obs}^j} \delta(r_{i,j} = v) + \eta}{\sum_l \sum_{r_{i,j} \in \mathcal{R}_{obs}^j} \delta(r_{i,j} = l) + \eta V}, \quad (2)$$

$$\tilde{S}(v) = \frac{\sum_{r_{i,j} \in \mathcal{R}_{obs}} \delta(r_{i,j} = v) + \eta}{\sum_l \sum_{r_{i,j} \in \mathcal{R}_{obs}} \delta(r_{i,j} = l) + \eta V}. \quad (3)$$

ここで、 $v, l = 1, 2, \dots, V$  であり、 $\tilde{P}_i(v)$  はユーザー  $i$  が評点  $v$  を付ける確率、 $\tilde{Q}_j(v)$  はアイテム  $j$  に対して

評点  $v$  が与えられる確率,  $\tilde{S}(v)$  は評点  $v$  の生起確率をそれぞれ表す.  $\eta$  は各評点の生起確率が 0 となることを防ぐためのスムーングパラメータを表し, 本論文では  $\eta = 1$  とした. また,  $\delta(\cdot)$  は引数が真のとき 1, それ以外のときは 0 を出力するインジケータ関数を表す. 明らかに  $\sum_v \tilde{P}_i(v) = \sum_v \tilde{Q}_j(v) = \sum_v \tilde{S}(v) = 1$ .

次に, 未評点箇所に対する上記 3 種類の (周辺) 評点分布,  $P_i (i = 1, 2, \dots, N), Q_j (j = 1, 2, \dots, M), S$ , が, 最初に求めた 3 種類の (周辺) 評点分布に類似するように未評点箇所の評点の組合せを求める (図 2 (b) 参照). なお, 各 (周辺) 評点分布,  $P_i, Q_j, S$  は, 式 (1) ~ (3) と同様の計算式でそれぞれ求める. たとえば,  $S$  は式 (3) 中の  $\mathcal{R}_{obs}$  を  $\mathcal{R}_{tar}$  で置き換えて計算する.  $P_i (i = 1, 2, \dots, N)$  と  $Q_j (j = 1, 2, \dots, M)$  も同様に置き換えてそれぞれ計算する.

各評点分布間の非類似性は, 以下で定義される, 2 つの分布間の偽距離を表す *Kullback-Leibler* (KL) ダイバージェンス<sup>9)</sup> によって測る.

$$\begin{aligned} \text{KL}(\tilde{P}_i || P_i) &= \sum_{v=1}^V \tilde{P}_i(v) \log \frac{\tilde{P}_i(v)}{P_i(v)}, \\ \text{KL}(\tilde{Q}_j || Q_j) &= \sum_{v=1}^V \tilde{Q}_j(v) \log \frac{\tilde{Q}_j(v)}{Q_j(v)}, \\ \text{KL}(\tilde{S} || S) &= \sum_{v=1}^V \tilde{S}(v) \log \frac{\tilde{S}(v)}{S(v)}. \end{aligned}$$

ここで, たとえば,  $\text{KL}(\tilde{P}_i || P_i)$  は評点分布  $\tilde{P}_i$  と評点分布  $P_i$  間の KL ダイバージェンスを表す. KL ダイバージェンスは非負の値をとり, 2 つの分布が一致するとき 0 となる.

結局, 評点予測問題は, 予測対象となる未評点の評点集合  $\mathcal{R}_{tar}$  を変数とする以下の目的関数  $J$ ,

$$\begin{aligned} J(\mathcal{R}_{tar}) &= \frac{1}{N} \sum_{i=1}^N \text{KL}(\tilde{P}_i || P_i) \\ &+ \frac{1}{M} \sum_{j=1}^M \text{KL}(\tilde{Q}_j || Q_j) \\ &+ \text{KL}(\tilde{S} || S), \end{aligned} \quad (4)$$

を最小化する最適化問題として定式化される. 式 (4) の左辺第 1 項の  $1/N$  と第 2 項の  $1/M$  は, ユーザ数やアイテム数の目的関数値への影響をなくすために導入している. 予測対象の評点を独立に予測する従来の CF 手法とは異なり, 提案法は, 予測対象の評点予測値を互いに依存させ, かつ, 一括して予測を行う.

### 3.3 予測アルゴリズム

目的関数  $J$  (式 (4)) を最小化する予測値の集合

$\{\hat{r}_{i,j} | \hat{r}_{i,j} \in \mathcal{R}_{tar}\}$  を求める方法として, 以下の最適化手法を適用することが考えられる.

#### (i) 組合せ最適化手法

評点を離散値と見なし, Particle Swarm Optimization<sup>8)</sup> などを適用することにより, 目的関数  $J$  を最小化する評点の組合せを求める.

#### (ii) 非線形最適化手法

評点を連続値と見なし, ニュートン法などを適用することにより, 評点が  $[1, V]$  であるという制約のもとで目的関数  $J$  を最小化する評点集合を求める.

しかし, (i) は, 予測対象の評点数  $\#\{\mathcal{R}_{tar}\}$  が大きくなるほど評点の組合せ数は膨大になるため, スケラビリティが重要視される CF においては, 実用的ではない. また (ii) は, 質の悪い局所解が多数存在することが分かり<sup>20)</sup>, (i) と同様に式 (4) を最小化するアプローチとしては適切ではない.

そこで本論文では, 実用性を重視し, 目的関数値が比較的小さい解を短時間で求めるヒューリスティックな予測アルゴリズムを提案する. この予測アルゴリズムは, (i) と同様に評点を離散値と見なし, 目的関数値が比較的小さい解からスタートして, 目的関数値が減少する限り評点を個別に更新し続ける. すなわち, 予測対象の全評点に関する最急勾配ではなく, 各評点に関する最急勾配を用いて徐々に目的関数を減少させることで, 質の悪い局所解に陥るのを防いでいる. 以下に詳細を示す.

---

予測アルゴリズム 入力:  $\mathcal{R}_{obs}, \mathcal{R}_{tar}$  出力:  $\mathcal{R}_{tar}$

---

初期化  $\mathcal{R}_{tar}$

for all  $r_{i,j} \in \mathcal{R}_{tar}$  do

以下を計算,

$$\alpha_{i,j}^P \leftarrow \arg \min_{r_{i,j}} \text{KL}(\tilde{P}_i || P_i),$$

$$\alpha_{i,j}^Q \leftarrow \arg \min_{r_{i,j}} \text{KL}(\tilde{Q}_j || Q_j),$$

$$\beta_{i,j}^{P+Q} \leftarrow \min_{r_{i,j}} \text{KL}(\tilde{P}_i || P_i) + \min_{r_{i,j}} \text{KL}(\tilde{Q}_j || Q_j).$$

end for

$\{r_{i,j} | r_{i,j} \in \mathcal{R}_{tar}\}$  を,  $\beta_{i,j}^{P+Q}$  の値で昇順にソート.

while 目的関数値  $J(\mathcal{R}_{tar})$  が減少 do

ソートした順に評点を更新,

$$r_{i,j} \leftarrow \alpha_{i,j}^P \quad \text{only if} \quad \alpha_{i,j}^P = \alpha_{i,j}^Q. \quad (5)$$

end while

---

実運用を考えた場合の, その手法の適用可能性を指す. 具体的には, 実装の容易性 (空間) 計算量, 計算時間など.

ここで,  $\arg \min_x f(x)$  は,  $f(x)$  を最小にする引数  $x$  を表す. また,  $\mathcal{R}_{tar}$  の初期化については, たとえば  $r_{i,j}$  を初期化する場合, ユーザ  $i$  の平均  $\bar{r}_i$  とアイテム  $j$  の平均  $\bar{r}_j$  の平均,

$$(\bar{r}_i + \bar{r}_j)/2, \quad (6)$$

とした. ユーザ  $i$  の平均  $\bar{r}_i$  とアイテム  $j$  の平均  $\bar{r}_j$  は以下によりそれぞれ計算する.

$$\bar{r}_i = \sum_{r_{i,j} \in \mathcal{R}_{obs}^i} r_{i,j} / \#\{\mathcal{R}_{obs}^i\},$$

$$\bar{r}_j = \sum_{r_{i,j} \in \mathcal{R}_{obs}^j} r_{i,j} / \#\{\mathcal{R}_{obs}^j\}.$$

ただし, 初期化するには整数値に丸めた値を用いる.

予測アルゴリズムにおいて,  $\beta_{i,j}^{P+Q}$  の値で予測対象の全評点をソートするのは, 目的関数値を大きく減少させる評点から優先的に評点更新を行うためである. ここで,  $\alpha_{i,j}^P$  ( $\alpha_{i,j}^Q$ ) は,  $r_{i,j}$  に関して目的関数の第 1 項 (第 2 項) を最も減少させる評点である. また, 式 (5) の評点更新によって目的関数の第 1 項と第 2 項の値が減少することになるが, 目的関数を減少させる評点値が一致している場合 ( $\alpha_{i,j}^P = \alpha_{i,j}^Q$ ) のみ評点を更新する. 一般に, 目的関数の第 1 項 (第 2 項) のみの最小化では, ユーザ間 (アイテム間) の関係が, 未評点の予測に反映されない. 式 (5) の制約は, ユーザ間, アイテム間の関係を考慮すべく設けた制約条件といえる. また, この制約により, 解空間での解探索の自由度を抑制し, 質の悪い局所解から回避できることも予備実験で確認している. たとえば, 評点  $r_{i,j}$  に関して, 第 1 項を最小化する値が 4 ( $\alpha_{i,j}^P = 4$ ), 第 2 項を最小化する値が 3 ( $\alpha_{i,j}^Q = 3$ ) であった場合には解は更新せず, 次に優先度の高い評点の更新に移る. 目的関数の第 3 項を考慮するため, 目的関数値が増加した時点で解の更新を停止する.

## 4. 評価実験

### 4.1 データセットと実験設定

CF におけるベンチマークデータとして広く用いられている映画の評価データである, *MovieLens* (ML1, ML2) と *EachMovie* (EM) の 3 種類の実データを用いて, 以降で説明する従来法と性能を比較した. ただし, EM に関しては, 評点数が 20 以上のユーザのみを用いた. 表 1 の上部に各データセットの詳細を示す.

実験設定は文献 12) と同様にした. つまり, 全ユーザを学習用ユーザとテストユーザに分け, 以下の 2 つの性能を評価した. 学習用ユーザとテストユーザのそ

表 1 MovieLens と EachMovie データセット  
Table 1 The MovieLens and EachMovie data sets.

	ML1	ML2	EM
ユーザ数	943	6,040	35,280
アイテム数	1,682	3,706	1,622
評点数	100,000	1,000,209	2,314,777
Sparsity	93.7%	95.5%	96.0%
評点のスケール	5	5	6
E[MAE]	1.6	1.6	1.944
学習用ユーザ数	800	5,000	30,000
テストユーザ数	143	1,040	5,280

れぞれの分割数を表 1 の下部に示す.

- 弱汎化性能: 既存ユーザに対する予測精度を評価する. 具体的には, 学習用ユーザの既知の評点を, 評点済みと未評点の 2 つに分け, 評点済みの評点を用いて未評点の評点を予測する. 評点予測の際, 学習用ユーザの評点済みの評点すべてを用い, テストユーザの評点はいっさい使用しない.
- 強汎化性能: 新規ユーザに対する予測精度を評価する. 具体的には, テストユーザの既知の評点を, 評点済みと未評点の 2 つに分け, 評点済みの評点を用いて未評点の評点を予測する. 評点予測の際, テストユーザの評点済みの評点と学習用ユーザの既知の評点を用いる. アクティブユーザ以外のテストユーザの評点済みの評点は使用しない.

また, 既知の評点を評点済みと未評点の 2 つに分ける方法として, いずれの評価においても AllBut  $n$  プロトコルを用いた. このプロトコルは, ユーザごとにランダムに  $n$  個の評点を未評点の評点として選択する. 実験では,  $n = 10\%$ ,  $20\%$  とし, それぞれ, “AllBut 10%”, “AllBut 20%” と表す.

### 4.2 従来法

提案法と比較を行う代表的な従来法を説明する. 提案法が予測対象のすべての評点を互いに依存させて一括予測するのに対し, 従来法は評点を個別に予測する.

#### 4.2.1 最近傍法に基づくアプローチ

先に説明したとおり, このアプローチは, 類似ユーザの評点履歴データベースから抽出し, 類似ユーザの評点済み評点を用いて予測を行う. アルゴリズムが単純なため実装が容易で, 実用化されている<sup>10),13)</sup>. ここで, 類似ユーザに基づいて予測を行う手法<sup>13)</sup> に対して, 最近では, 類似アイテムに基づいて予測を行う手法<sup>4),10),14)</sup> も提案されている. 後者は, 評点行列  $R$  を転置した行列に対して, 従来どおりの最近傍法に基づく予測手法を適用する手法である. 以降, 類似ユーザ (類似アイテム) に基づく手法を user-base (item-base) と呼ぶ. 通常, ユーザ数  $N$  よりもアイテム数

$M$  の方が少ないため、類似ユーザを探索するよりも類似アイテムを探索する方が計算時間の点で優れている。様々な類似性を測る尺度が提案されているが、予備実験をふまえ、今回の実験では Pearson 相関係数<sup>13)</sup> と Adjusted Cosine 類似度<sup>14)</sup> を採用することとした。以下、このアプローチに基づく 2 つの従来法について説明する。

- $k$  近傍法 ( $k$ NNs)<sup>10),13),14)</sup>

何らかの類似尺度を用いて抽出した  $k$  人 (個) の類似ユーザ (アイテム) のターゲットアイテム (アクティブユーザ) に対する評点を、類似度によって重み付けした値により予測を行う。具体的には以下の式により予測値を求める。

$$\begin{aligned} \text{user-base : } \hat{r}_{a,t} &= \frac{\sum_{i=1}^k W_a^i (r_{i,t} - \bar{r}_i)}{\sum_{i=1}^k |W_a^i|} + \bar{r}_a, \\ \text{item-base : } \hat{r}_{a,t} &= \frac{\sum_{j=1}^k W_t^j (r_{a,j} - \bar{r}_j)}{\sum_{j=1}^k |W_t^j|} + \bar{r}_t. \end{aligned}$$

ここで、 $W_a^i$  はアクティブユーザ  $a$  とユーザ  $i$  の類似度を表す。同様に、 $W_t^j$  はターゲットアイテム  $t$  とアイテム  $j$  の類似度を表す。

$k$ NNs を適用する際には、使用するデータに依存して、類似性を測る対象、類似尺度、近傍数  $k$  を決定する必要がある。実験では、user-base/item-base、Pearson 相関係数/Adjusted Cosine 類似度、 $k = 100, 200, \dots, 500$  の中で最も予測精度が良い組合せを  $k$ NNs の性能とした。

- Unified method with Similarity Fusion (SF)<sup>16)</sup>

user-base  $k$ NNs と item-base  $k$ NNs を統合、拡張した手法である。類似ユーザと類似アイテムからなる評点行列に存在するすべての評点済みの評点 ( $\mathcal{R}_{nns}$  とおく) と、個々の評点に対する重みを用いた重み付き平均により予測する。そのため、類似ユーザのターゲットアイテムに対する評点や、類似アイテムのアクティブユーザに対する評点だけでなく、類似ユーザの類似アイテムに対する評点も予測式に取り込まれることになる。予測式は以下で表される。

$$\hat{r}_{a,t} = \sum_{r_{i,j} \in \mathcal{R}_{nns}} W_{a,t}^{i,j} f_{a,t}(r_{i,j}), \quad (7)$$

$$f_{a,t}(r_{i,j}) = r_{i,j} - (\bar{r}_i - \bar{r}_a) - (\bar{r}_j - \bar{r}_t).$$

ここで、 $W_{a,t}^{i,j}$  は  $r_{i,j}$  の  $r_{a,t}$  に対する重みを表す。

SF は、1 つの評点を予測するごとに、類似ユーザと類似アイテムの抽出を行う必要がある。実験では、文献 16) と同様の設定を用いた (たとえば、ユーザ間類似度は Cosine 類似度、アイテム間類似度は Adjusted

Cosine 類似度を用いた)。詳細は文献 16) 参照。

#### 4.2.2 確率モデルに基づくアプローチ

上述のとおり、このアプローチは、各評点が生成される確率モデルを仮定し、過去のデータをもとに学習したモデルを用いて、将来の評点を予測する<sup>6),11),12),17),19)</sup>。予測対象の評点が離散値 ( $v = 1, 2, \dots, V$ ) であることから、各評点が従う分布として多項分布が仮定される。評点を予測する際には、以下で定義される *median rating*<sup>12)</sup>,

$$\left\{ v \left| \Pr\{r_{a,t} < v\} \leq \frac{1}{2}, \frac{1}{2} \leq \Pr\{r_{a,t} > v\} \right. \right\},$$

を  $r_{a,t}$  の予測値  $\hat{r}_{a,t}$  として用いる。ここで、

$$\sum_{v=1}^V v \Pr\{r_{a,t} = v\},$$

により計算される評点の平均値によって予測を行うことも考えられるが、予備実験の結果、予測精度が良かった *median rating* を採用した。以下、このアプローチに基づく 4 手法について説明する。

- 多項分布モデル (MULTI)<sup>2)</sup>

ユーザに依存せず、各アイテムの評点分布に対して 1 つの多項分布を仮定するモデル。多項分布のパラメータ (各評点の出現確率) は式 (2) により計算する。アイテム  $j$  の評点が従う多項分布を  $p(r_{i,j}|j)$  で表すと、ユーザ  $i$  の評点が従う結合評点分布は以下で表される。

$$p(r_{i,1}, r_{i,2}, \dots, r_{i,M}) = \prod_{j=1}^M p(r_{i,j}|j)^{\delta(r_{i,j} \neq 0)}.$$

- 混合多項分布モデル (MIXMULTI)<sup>12)</sup>

ユーザはいくつかの嗜好の似通ったグループ (潜在クラス) に分けられると仮定するモデル。つまり、同じ潜在クラスに属するユーザの評点は、アイテムごとに同一の多項分布に従うと仮定する。ユーザ  $i$  の評点が従う結合評点分布は以下で表される。

$$\begin{aligned} p(r_{i,1}, r_{i,2}, \dots, r_{i,M}) \\ = \sum_{c=1}^C p(z_c) \prod_{j=1}^M p(r_{i,j}|j, z_c)^{\delta(r_{i,j} \neq 0)}. \end{aligned}$$

ここで、 $p(z_c)$  はクラス  $z_c$  が生起する確率、 $p(r_{i,j}|j, z_c)$  はクラス  $z_c$  におけるアイテム  $j$  の評点が従う多項分布をそれぞれ表す。明らかに予測精度はクラス数  $C$  に依存する。実験では、 $L = 5, 10, 15$  として最も予測精度の良い値を MIXMULTI の性能とした。ここで、モデルに含まれる未知パラメータの学習には Expectation-Maximization (EM) アルゴリズム<sup>3),12)</sup> を用い、過学習を防ぐため反復回数は最

大 100 回とした。

• Aspect Model (AM)<sup>(6),12)</sup>

ユーザは  $C$  個の潜在クラスに分かれ、同じ潜在クラスに属するユーザの評点は、アイテムごとに同一の多項分布に従うと仮定するモデル。MIXMULTI と異なる点は、AM ではアイテムごとにユーザが属するクラスが異なること（ユーザが複数のクラスに属すること）を許している点である。ユーザ  $i$  の評点が従う結合評点分布は以下で表される。

$$p(r_{i,1}, r_{i,2}, \dots, r_{i,M}) = \prod_{j=1}^M \sum_{c=1}^C p(z_c|i) p(r_{i,j}|j, z_c)^{\delta(r_{i,j} \neq 0)}. \quad (8)$$

ここで、AM は潜在クラスの出現確率をユーザで条件付けるため、新規ユーザに対して予測を行う場合にはモデルの再学習が必要であること（生成モデルではないこと）に注意。実験では、MIXMULTI と同様に  $L = 5, 10, 15$ 、学習の反復回数は最大 100 回とした。このモデルは情報検索の分野で考案された pLSA<sup>7)</sup> を CF に適用したモデルである。

• User Rating Profile Model (URP)<sup>(11),12)</sup>

式 (8) 中の  $p(z_c|i)$  を  $p(z_c|\theta)p(\theta; \gamma)$  で置き換えることで AM を拡張し、厳密に生成モデル化したモデル。ここで、 $p(\theta; \gamma)$  は  $\gamma$  をハイパーパラメータとする Dirichlet 分布を表す。ユーザ  $i$  の評点が従う結合評点分布は以下で表される。

$$p(r_{i,1}, r_{i,2}, \dots, r_{i,M}; \gamma) = \int_{\theta} p(\theta; \gamma) \prod_{j=1}^M \sum_{c=1}^C p(z_c|\theta) p(r_{i,j}|j, z_c)^{\delta(r_{i,j} \neq 0)} d\theta.$$

URP は AM のように新規ユーザに対する予測を行うたびにモデルを再学習する必要はないが、推定すべきパラメータ数が増え、学習は AM に比べて複雑になる。実験では、MIXMULTI や AM と同様に  $L = 5, 10, 15$  とした。ただし、URP の学習は局所解が多く困難なため、文献 12) と同様に初期値として MIXMULTI の結果を用い、学習の反復回数は最大 10 回とした。

### 4.3 結果と考察

Dual Xeon 3.60 GHz CPU、2 GB メモリの計算機で実験した結果を表 2 と表 3 に示す。表 2 は予測精度、表 3 は予測に要した計算時間である。ここで、表中の BASE LINE は、式 (6) による予測精度を示している。式 (6) は容易に計算でき、かつ、比較的良い予測精度が得られるためベースラインとした。ただし、BASE LINE では値の丸め操作は行わない。表中の“\*” は、計算時間が 2 週間 (336 h) 以上かかった

め、計算を中止させたことを示す。また、AM は新規ユーザが追加されるたびにモデルの再学習が必要となるため、新規ユーザに対して予測を行う強化性能評価実験への適用は適切ではない。よって、これに該当する箇所を“\*”で示している。ここで、表 3 中の h, m, s はそれぞれ時間, 分, 秒を表す。表中の各値は、学習用ユーザとテストユーザの分割を各データセットごとに 3 パターン作成し、さらに、各パターンに対して評点済みと未評点の分割を 3 パターン作成して得られたデータに対する結果の平均値を表す。

#### 4.3.1 予測精度に関する考察

表 2 に示すとおり、(a) と (b) のいずれの設定においても、すべての手法は BASE LINE よりも予測精度が良い。また、ML1 と EM の NMAE を比較することで、データの規模が大きくなるほど各手法の予測精度が向上することが確認できる。データの規模拡大にともなう予測精度の向上は、確率モデルに基づくアプローチ (MIXMULTI, AM) で顕著である。これは、規模が大きいくほど情報が多く、ユーザは嗜好性に基づいてクラスタ化されるという直感的な理解と一致する。

提案法は、データの規模によらず安定的な値を示しており、特にデータの規模が小さいときに最も良い性能を示している。これはデータの規模が小さく、評点に関する全体的な情報が少ない場合には、予測値を互いに依存させることで、予測精度が向上することを示している。逆に、データの規模が大きくなると、提案法の性能は他手法に劣る。特に、確率モデルに基づく手法 (MIXMULTI, AM) との差が大きく、大規模データに対しては、潜在クラス導入の効果があるといえる。ただし、MIXMULTI, AM, URP は潜在クラス数  $C$  を、データに合わせて手作業で決定する必要があること、つまり、実験設定したクラス数の中で最も良い性能と比較していることに注意。SF は kNNs よりも悪い結果が得られた。クラス数を探索的に決定する必要がある MIXMULTI などと同様に、データに合わせてパラメータを設定する必要がある。

いずれの手法も同等の予測精度を示しており、かつ、データによって各手法の優劣は変化しているため、予測精度の観点で最も優れている手法は決定できない。

#### 4.3.2 計算時間に関する考察

表 3 に示すとおり、データの規模が大きくなるにつれて計算時間は多くなる。しかし、データの規模拡大にともなう計算時間の増加率は手法によって大きく異なっている。提案法は、他手法に比べてかなり高速であることが確認できる。ここで、MULTI や BASE

表 2 各手法の予測精度．値が小さいほど予測精度が良いことを意味する

Table 2 The prediction performance of each method. A smaller value means better performance.

(a) 弱汎化性能

		提案法	kNNs	SF	MULTI	MIXMULTI	AM	URP	BASE LINE
ML1	AllBut10%	<b>0.456</b>	0.468	0.477	0.488	0.489	0.470	0.487	0.498
	AllBut20%	<b>0.457</b>	0.470	0.479	0.490	0.492	0.486	0.492	0.499
ML2	AllBut10%	0.443	0.440	0.459	0.464	0.445	<b>0.416</b>	0.464	0.481
	AllBut20%	0.446	0.441	0.459	0.465	0.444	<b>0.419</b>	0.468	0.482
EM	AllBut10%	0.445	0.419	0.451	0.463	0.409	<b>0.403</b>	0.465	0.469
	AllBut20%	0.447	0.421	*	0.463	0.411	<b>0.405</b>	0.465	0.469

(b) 強汎化性能

		提案法	kNNs	SF	MULTI	MIXMULTI	AM	URP	BASE LINE
ML1	AllBut10%	<b>0.450</b>	0.456	0.463	0.477	0.464	-	0.474	0.488
	AllBut20%	<b>0.458</b>	0.461	0.469	0.480	0.476	-	0.486	0.493
ML2	AllBut10%	<b>0.446</b>	0.449	0.460	0.467	0.448	-	0.472	0.483
	AllBut20%	0.447	0.452	0.461	0.468	<b>0.446</b>	-	0.470	0.484
EM	AllBut10%	0.445	0.428	0.456	0.462	<b>0.407</b>	-	0.465	0.469
	AllBut20%	0.446	0.430	0.451	0.462	<b>0.406</b>	-	0.464	0.469

表 3 各手法の予測に要した計算時間．要した時間が短いほどスケーラビリティが良いことを意味する

Table 3 The computational time of each method. A smaller value means better scalability.

(a) 弱汎化性能

		提案法	kNNs	SF	MULTI	MIXMULTI	AM	URP	BASE LINE
ML1	AllBut10%	0.67 s	1.24 m	5.10 m	<b>0.22 s</b>	11.0 s	5.74 m	4.42 m	<b>0.22 s</b>
	AllBut20%	1.00 s	2.19 m	8.67 m	<b>0.21 s</b>	8.33 s	6.40 m	4.39 m	0.22 s
ML2	AllBut10%	11.9 s	2.36 h	22.4 h	<b>4.22 s</b>	10.3 m	1.65 h	1.20 h	4.33 s
	AllBut20%	15.0 s	3.82 h	40.4 h	<b>4.11 s</b>	12.8 m	1.67 h	4.00 h	4.22 s
EM	AllBut10%	41.3 s	12.2 h	220 h	<b>13.9 s</b>	3.98 h	5.35 h	3.54 h	14.1 s
	AllBut20%	50.6 s	21.5 h	*	<b>13.8 s</b>	4.71 h	5.48 h	3.18 h	14.2 s

(b) 強汎化性能

		提案法	kNNs	SF	MULTI	MIXMULTI	AM	URP	BASE LINE
ML1	AllBut10%	0.44 s	15.8 s	1.01 m	<b>0.18 s</b>	17.0 s	-	11.3 m	0.22 s
	AllBut20%	0.56 s	31.3 s	1.95 m	<b>0.18 s</b>	17.7 s	-	10.2 m	0.11 s
ML2	AllBut10%	8.44 s	11.1 m	4.57 h	<b>3.78 s</b>	11.4 m	-	1.48 h	4.08 s
	AllBut20%	9.11 s	18.9 m	7.42 h	<b>3.89 s</b>	15.1 m	-	1.66 h	3.96 s
EM	AllBut10%	29.7 s	1.21 h	59.9 h	<b>12.8 s</b>	4.73 h	-	5.13 h	13.8 s
	AllBut20%	31.8 s	1.62 h	145 h	<b>12.8 s</b>	4.20 h	-	4.20 h	13.4 s

LINE は提案法よりもさらに高速に予測できるが、その差は、それら以外の手法の計算時間との差に比べれば大差ではない。また、MULTI や BASE LINE は、予測性能面で、全データにおいて提案法よりも劣っている。

提案法とは対照的に、kNNs, SF, MIXMULTI, AM, URP は、データの規模に合わせて急激に計算時間が増大している。ここで、SF の計算時間が他と比べて特に多いのは、類似ユーザの探索と類似アイテムの探索を 1 台の計算機で行っているためであり、並列処理により改善は可能である。

各手法の計算量を表 4、空間計算量を表 5 にそれぞれ示す<sup>12)</sup>。表中の  $T$  は予測対象の評点の総数  $\#\{r_{tar}\}$  を表し、 $I, I_1, I_2$  は EM アルゴリズムにおける反

復回数をそれぞれ表す。また、予測対象の評点が決定される以前にあらかじめ計算できる量を学習に要する（空間）計算量、予測対象の評点が決定された後に計算する量を予測に要する（空間）計算量としている。ただし、kNNs に関して、弱汎化性能評価実験では類似ユーザ（アイテム）をあらかじめ計算しておくことが可能だが、強汎化性能評価実験では不可能ゆえ、予測に要する計算量とした。表 4 と表 5 から、最近傍法に基づく手法は予測の際に（空間）計算量がかかり、確率モデルに基づく手法は学習の際に（空間）計算量がかかることが分かる。ここで、提案法と従来法との計算量を比較してみると、最近傍法に基づくアプローチについては  $NMT \gg TV$  であること、確率モデルに基づくアプローチについては  $NMVICI \gg NMV$



表 4 予測対象の全評点の予測に要する計算量 (computational complexity)  
Table 4 The computational complexity to predict all target ratings.

	学習に要する計算量	予測に要する計算量
提案法	$O(NMV)$	$O(TV + T \log T + T)$
$k$ NNs(user-base)	-	$O(NMT + NT \log N + kT)$
$k$ NNs(item-base)	-	$O(NMT + MT \log M + kT)$
SF	-	$O(NMT + NT \log N + MT \log M + k^2T)$
Multi	$O(NMV)$	$O(T)$
MixMulti	$O(NMVCI)$	$O(MVCT)$
AM	$O(NMVCI)$	$O(MVCT)$
URP	$O(NMVCI_1I_2)$	$O(MVCI_1T)$
BASE LINE	$O(NM)$	$O(T)$

$T$ : 予測対象の評点の総数  $\#\{\mathcal{R}_{tar}\}$ .  $I, I_1, I_2$ : 反復回数.

表 5 予測に要する空間計算量 (space complexity)  
Table 5 The space complexity to predict all target ratings.

	学習に要する空間計算量	予測に要する空間計算量
提案法	$O(V)$	$O(NV + MV)$
$k$ NNs(user-base)	-	$O(NM)$
$k$ NNs(item-base)	-	$O(NM)$
SF	-	$O(NM)$
Multi	$O(V)$	$O(M)$
MixMulti	$O(NC + MVC + C)$	$O(MVC + C)$
AM	$O(NMVC + MVC + NC)$	$O(MVC + NC)$
URP	$O(NMC + NC + MVC + C)$	$O(MVC + C)$
BASE LINE	$O(1)$	$O(NM)$

$T$ : 予測対象の評点の総数  $\#\{\mathcal{R}_{tar}\}$ .

であることから、提案法が従来法に比べて短時間で予測できることが分かる。また、予測対象の評点の総数  $T$  はユーザ数  $N$  やアイテム数  $M$  に直接的には影響を受けず、本論文においては  $\#\{\mathcal{R}_{tar}\} \approx \#\{\mathcal{R}_{mis}\}$  であるため、提案法は短時間で予測が行えている。

#### 4.3.3 考察のまとめ

実世界で扱われているデータの規模は、今回の実験で用いたデータの規模に比べはるかに大きい (数十、百万ユーザ/アイテム)。したがって、CF においてスケラビリティは特に重要であり、単純に予測精度を比較するだけでは各手法の優劣を判断することはできない。このような背景をふまえ、提案法は、従来法と同程度の予測精度を短時間で得られるという点で実用上の価値があるといえる。

## 5. まとめ

予測対象の評点を互いに依存させ、全評点を一括して予測する協調フィルタリング手法を提案した。評点を個別に予測する従来法とは異なり、提案法は、予測対象の評点以外の評点に対する予測値も考慮した予測手法となっている。性能評価実験の結果、提案法は従来法と比べて大幅に短い計算時間で、従来法とほぼ同等の予測精度を得ることを確認した。提案法は MAR 仮定のみに基づいているため、CF に限らず様々な欠損

データにも適用可能である。今後は、超大規模なデータ (数百万ユーザ) を対象に、さらに効果的な CF 手法を検討する予定である。

謝辞 本研究の機会を与えていただいた、株式会社 NTT データ技術開発本部ソフトウェア工学推進センタ木谷強センタ長、同ビジネスインテリジェンス推進センタ島崎勝美部長、松永務課長、ならびに、当研究所協創情報研究部創発環境研究グループ山田武士グループリーダーに深く感謝いたします。

## 参考文献

- 1) Adomavicius, G. and Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, *IEEE Trans. Knowledge and Data Engineering*, Vol.17, No.6, pp.734–749 (2005).
- 2) Breese, J.S., Heckerman, D. and Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering, *Proc. 14th Conference on Uncertainty in Artificial Intelligence (UAI)*, Madison, Wisconsin, US, pp.43–52 (1998).
- 3) Dempster, A.P., Laird, N.M. and Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society Series B*, Vol.34, pp.1–38 (1977).

- 4) Deshpande, M. and Karypis, G.: Item-Based Top-N Recommendation Algorithms, *ACM Trans. Information Systems*, Vol.22, No.1, pp.143–177 (2004).
- 5) Getoor, L. and Sahami, M.: Using Probabilistic Relational Models for Collaborative Filtering, *Proc. Workshop on Web Usage Analysis and User Profiling (WEBKDD-99)*, San Diego, CA, US (1999).
- 6) Hofmann, T.: Latent Semantic Models for Collaborative Filtering, *ACM Trans. Information Systems*, Vol.22, No.1, pp.89–115 (2004).
- 7) Hofmann, T.: Unsupervised Learning by Probabilistic Latent Semantic Analysis, *Machine Learning*, Vol.42, pp.177–196 (2004).
- 8) Kennedy, J. and Eberhart, R.: Particle swarm optimization, *Proc. IEEE International Conference on Neural Networks*, Piscataway, NJ, US, pp.1942–1948 (1995).
- 9) Kullback, S. and Leibler, R.: On information and sufficiency, *Annals of Mathematical Statistics*, Vol.22, pp.79–86 (1951).
- 10) Linden, G., Smith, B. and York, J.: Amazon.com Recommendations: Item-to-Item Collaborative Filtering, *IEEE Internet Computing*, Vol.7, No.1, pp.76–80 (2003).
- 11) Marlin, B.: Modeling User Rating Profiles For Collaborative Filtering, *Proc. 17th Annual Conference on Neural Information Processing Systems (NIPS-2003)*, British Columbia, Canada (2001).
- 12) Marlin, B.: Collaborative Filtering: A Machine Learning Perspective, Master's thesis, University of Toronto (2004).
- 13) Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J.: GroupLens: An Open Architecture for Collaborative Filtering of Netnews, *Proc. ACM CSCW1994*, pp.175–186 (1994).
- 14) Sarwar, B., Karypis, G., Konstan, J. and Riedl, J.: Item-Based Collaborative Filtering Recommendation Algorithms, *Proc. ACM WWW2001*, pp.285–295 (2001).
- 15) Shardanand, U. and Maes, P.: Social Information Filtering: Algorithms for Automating “Word of Mouth”, *Proc. ACM CHI1995*, Denver, Colorado, US, pp.210–217 (1995).
- 16) Wang, J., de Vries, A.P. and Reinders, M.J.: Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion, *Proc. ACM SIGIR2006* (2006).
- 17) Xu, Z., Tresp, V., Yu, K. and Kriegel, H.-P.: Infinite Hidden Relational Models, *Proc. 22nd International Conference on Uncertainty in Artificial Intelligence (UAI2006)*, Cambridge, MA, US (2006).
- 18) Yu, K., Tresp, V. and Yu, S.: A Nonparametric Hierarchical Bayesian Framework for Information Filtering, *Proc. ACM SIGIR2004*, Sheffield, UK (2004).
- 19) Zitnick, C.L. and Kanade, T.: Maximum Entropy for Collaborative Filtering, *Proc. 20th Conference on Uncertainty in Artificial Intelligence (UAI)*, Edinburgh, Scotland, pp.636–643 (2004).
- 20) 桑田修平, 上田修功: 周辺評点分布に基づく協調フィルタリング手法, *信学技報*, Vol.AI2006-03, pp.13–18 (2006).

(平成 19 年 2 月 2 日受付)

(平成 19 年 3 月 22 日再受付)

(平成 19 年 4 月 22 日採録)



桑田 修平 (正会員)

1977 年生 . 2001 年早稲田大学理工学部経営システム工学科卒業 . 2003 年同大学大学院修士課程修了 . 同年株式会社 NTT データ入社 . 現在 , 日本電信電話株式会社 NTT コミュニケーション科学基礎研究所に所属 . 統計的学習 , データマイニング等の研究に従事 . 電子情報通信学会 , 人工知能学会各会員 .



上田 修功 (正会員)

1958 年生 . 1982 年大阪大学工学部通信工学科卒業 . 1984 年同大学大学院修士課程修了 . 工学博士 . 同年 NTT 入社 . 1993 年より 1 年間 Purdue 大学客員研究員 . 画像処理 , パターン認識・学習 , ニューラルネットワーク , 統計的学習 , Web データマイニング等の研究に従事 . 現在 , NTT コミュニケーション科学基礎研究所協創情報研究部長 . 奈良先端科学技術大学院大学客員教授 . 電気通信普及財団賞 (テレコムシステム技術賞) (1997 年 , 2006 年) , 電子情報通信学会論文賞 (2000 年 , 2004 年) , 情報処理学会山下記念研究賞 (2006 年) 等受賞 . 電子情報通信学会 , 日本神経回路学会 , IEEE 各会員 .