

推薦システムのアルゴリズム

Algorithms of Recommender Systems

神島 敏弘 <<http://www.kamishima.net/>>

Release: 2016-09-26 21:53:16 +0900; 9645c3b

まえがき

本稿は推薦システムについてまとめたものである。

人工知能学会誌 2007 年 11 月号 [神寫 07], 2008 年 1 月号 [神寫 08a], および 2008 年 3 月号 [神寫 08b] の 3 回に渡って連載した解説記事「推薦システムのアルゴリズム (1)~(3)」に対し、誤りの訂正や、新しい内容の追加などの更新を行ったものである。

本稿のソースファイルは `GitHub` にて公開している。

<https://github.com/tkamishima/recsysdoc>

TYPO や記述の誤りなどのバグリポートは、`GitHub` の `pull request` か、`issues` を使って連絡されたい。なお、事情によりすぐには対処できない場合があるので、予めご了解いただきたい。

本稿の構成

本稿の構成は以下のとおりである。第 I 部では、推薦システムとは何か、またその設計指針や分類について述べる。第 II 部では、データの入力、嗜好の予測、そして推薦の提示からなる推薦システムの実行過程について述べる。第 III 部では、さまざまな嗜好の予測アルゴリズムのを紹介する。第 IV 部では、推薦システムに関連する話題や、さまざまな状況での推薦を紹介する。第 V 部は関連資料の紹介とまとめである。

謝辞

チュートリアル記事の執筆にあたり以下の方々の協力を得たことに感謝する。麻生英樹，岩田具治，佐藤健，廣瀬勝一，藤井敦，村上知子，山口高平には，本稿に関する貴重なコメントをいただいた。J. Riedl, J. Herlocker には論文の詳細について教えていただいた。アマゾンジャパン様，アップルジャパン様，GroupLens プロジェクト様には WWW のスクリーンショットなどの掲載を許可いただいた。

本稿の更新に関して次の方々の協力を得たことに感謝する：赤穂昭太郎，石黒勝彦，岡野原大輔，奥健太，折田明子，酒井哲也，佐久間淳，佐藤一誠，富岡亮太，中川裕志，土方嘉徳，星野伸明

数式の表記

スカラーの変数はイタリック体 x で、一部の確率変数は大文字のイタリック体 X 、ベクトルは小文字ボールド体 \mathbf{x} で、行列は大文字ボールド体 \mathbf{X} で表記する。実数などの特殊なものを除き、集合にはカリグラフィック体 \mathcal{D} を用いる。

表記	意味	表記	意味
x	特定のユーザーを表す	y	特定のアイテムを表す
X	ユーザーを表す確率変数	Y	アイテムを表す確率変数
\mathbf{x}	ユーザーをまとめたベクトル	\mathbf{y}	アイテムをまとめたベクトル
n	利用者数	m	アイテム数
\mathcal{X}	利用者集合 $\{1, \dots, n\}$	\mathcal{Y}	アイテム集合 $\{1, \dots, m\}$
\mathcal{X}_y	アイテム y を評価した利用者の集合	\mathcal{Y}_x	利用者 x が評価したアイテムの集合
a	活動ユーザーを表す	r_{xy}	利用者 x のアイテム y への評価値
\bar{r}_x	利用者 x による評価値の平均	\bar{r}_y	アイテム y への評価値の平均
\mathbf{R}	評価値行列	\mathcal{R}	評価値集合 (5 段階評価なら $\{1, \dots, 5\}$)
\mathbf{r}	評価値をまとめたベクトル	z	潜在因子
\mathbf{z}	潜在因子のベクトル	K	潜在因子の数・次元数
\mathcal{D}	データ集合	N	訓練データ数
\mathbf{U}	利用者潜在因子行列	\mathbf{V}	アイテム潜在因子行列
\mathbf{u}_x	利用者 x の潜在因子ベクトル	\mathbf{v}_y	アイテム y の潜在因子ベクトル
$y^{(t)}$	時刻 t での値	$\langle Y^{(t)} \rangle$	アイテムの時系列
$\theta \Theta$	パラメータを一般に表す	sig()	シグモイド関数
Dom()	変数の定義域	\perp	欠損値

スカラー関数 $f(x)$ に対して、その引数をベクトルとする表記 $f(\mathbf{x})$ は、ベクトル \mathbf{x} の各要素を関数 f に適用して得られるベクトルを表す。

確率変数 X が離散の場合の確率質量関数も、連続値の場合の確率密度関数も特に区

別することなく $\Pr[X]$ と表記する.

$E_{\Pr[X]}[f(X)]$ は, 分布 $\Pr[X]$ についての次の期待値を表す:

$$\begin{aligned} \sum_{x \in \text{Dom}(X)} f(x) \Pr[X = x] & \dots\dots X \text{ が離散の場合} \\ \int_{x \in \text{Dom}(X)} f(x) \Pr[X = x] dx & \dots\dots X \text{ が連続の場合} \end{aligned}$$

なお, $\Pr[X]$ を省略した場合は, 関数 f の全ての確率変数の同時分布に関する期待値を表す. 例えば, $E[f(X, Y)]$ は, $E_{\Pr[X, Y]}[f(X, Y)]$ の意味である.

目次

第 I 部	推薦システムの概要	1
第 1 章	推薦システム	2
第 2 章	推薦システムの分類と目的	5
2.1	推薦の個人化の度合い	6
2.2	推薦システムの運用目的の分類	8
2.3	推薦システムの予測タスクの分類	10
2.4	推薦システムの利用動機の分類	11
第 3 章	推薦システム設計の要素	13
3.1	推薦の性質	13
3.2	推薦候補の予測に関する制約	19
第 II 部	推薦システムの実行過程	21
第 4 章	推薦システムの実行過程	22
第 5 章	データの入力	24
5.1	暗黙的と明示的な嗜好データの獲得	25
5.2	明示的な獲得	26

5.3	暗黙的な獲得	31
5.4	嗜好データのその他の要因	32
5.5	嗜好データ以外のデータ	32
第 6 章	嗜好の予測	34
6.1	内容ベースと協調フィルタリングの比較	38
第 7 章	推薦の提示	43
7.1	推薦の配送	43
7.2	推薦アイテムの選別	44
7.3	推薦の表示形式	45
7.4	多様性の向上	46
7.5	推薦理由	46
第 III 部	推薦システムのアルゴリズム	49
第 8 章	協調フィルタリング	50
8.1	メモリベース法とモデルベース法	50
第 9 章	メモリベース型協調フィルタリング	53
9.1	利用者間型メモリベース法	53
9.2	アイテム間型メモリベース法	59
9.3	メモリベース法に関するその他の研究	60
第 10 章	協調フィルタリング：モデルベース法	62
10.1	クラスタモデル	62
10.2	関数モデル	63
10.3	確率モデル	68
10.4	時系列モデル	75

目次	vii
第 11 章 協調フィルタリング：ハイブリッド	81
第 12 章 内容ベースフィルタリング	83
12.1 入力の形式	83
12.2 推薦規則の獲得	84
第 13 章 嗜好の予測：まとめ	86
13.1 ハイブリッド法	86
13.2 嗜好の予測手法の選択	92
第 IV 部 推薦システムのその他の話題	95
第 14 章 プライバシー保護協調フィルタリング	96
14.1 暗号化による方法	97
14.2 ランダム化による方法	99
第 15 章 サクラ攻撃	100
15.1 サクラ攻撃の基本	101
15.2 サクラ攻撃の種類	102
15.3 サクラ攻撃に対する防御	105
第 16 章 推薦システムのその他の問題や視点	107
16.1 対話型システム	107
16.2 標本利用者の選別	110
16.3 推薦システムの運用	110
第 V 部 まとめ	114
第 17 章 参考資料の紹介	115

17.1	代表的な文献, 解説記事, および書籍	115
17.2	チュートリアル講演	117
17.3	Web サイト	117
17.4	テスト用データ	118
17.5	ソフトウェア	119
 参考文献		 120
 索引		 133

第 I 部

推薦システムの概要

第 1 章

推薦システム

最初に『**推薦システム** (recommender system)』とは何であるかということについて、その原点の一つである ACM Communications 誌での特集 [Resnick 97] での記述を紹介する：

It is often necessary to make choices without sufficient personal experience of the alternatives. In everyday life, we rely on recommendations from other people either by word of mouth, recommendation letters, movie and book reviews printed in newspapers, or general surveys such as Zagat's restaurant guides. Recommender systems assist and augment this natural social process.

自分の経験だけでは違いがあまりよくわからないものの中からでも、どうしてもどれかを選ばなければならないということはよくある。こうしたときには、口コミ、推薦状、新聞の書評や映画評、ザガットのレストランガイドなどの他人からの推薦に頼ることを日常的に行っている。
推薦システムは、こうした社会で普通に行われている一連の行為を補助したり、促進したりする。

より簡潔には、Konstan によるチュートリアル [Konstan 03] の定義がよいだろう：

Recommenders: Tools to help identify worthwhile stuff

推薦システム：どれに価値があるかを特定するのを助ける道具

このように、利用者にとって有用と思われる対象、情報、または商品などを選び出し、それを利用者の目的に合わせた形で提示するシステムといえる。

この推薦システムが必要になった背景は大きく二つある。第一に、大量の情報が発信されるようになったことがある。これは、情報化技術の進展により、個人・団体が容易かつ低コストで発信できるようになったためである。第二の理由は、これら大量の情報の蓄積や流通が容易になり、誰もが大量の情報を得ることができるようになったことである。これも計算機の記憶媒体の大規模化や、通信の高速化によるものである。これらの要因により、大量に発信された情報を、だれもが大量に取得できる状況が生じた。しかし、どのように欲しい情報を特定する方法が分からない（例：統計資料として公開されているがその名前が分からない）とか、探している情報を特定できない（例：類似した資料が大量にあり、その中に目的のものが埋もれてしまっている）といった理由により、情報を参照できる状態にあるにもかかわらず、それを識別できないという状況が生じた。この状況を『**情報過多 (information overload)**』 [Maes 94] (情報爆発 (information explosion) や情報洪水 (information overflow)) という。この状況に対処するため、利用者にとって有用な情報を見つけ出す推薦システムは考案された。

この推薦システムがどのように誕生し、広まっていったかを述べておく。広義には情報検索や情報フィルタリング技術の一つと見なせるので、初期の推薦システムはこれらの技術を基盤としていた。この推薦システムの実現手法の一つに協調フィルタリングがあるが、この用語の方が推薦システムという用語より古く、1992年に文献 [Goldberg 92] にて使われた。しかし、これは現在のような協調フィルタリングではなく、他人が手動で行った推薦を検索できる協調作業支援のシステムであった。この過程を自動化したシステムが1994年の GroupLens [Resnick 94] や Ringo [Shardanand 95] であり、現在の推薦システムの基礎となった。同時に、もう一つの実現手法である内容ベースフィルタリングも、従来からある情報フィルタリングとして、また、事例ベース推論の応用としても研究されてきたが、推薦システムとして独自の側面が徐々に強くなっていった。1996年には、専門のワークショップ

も開催されるほどに研究が活発化した。1997年には上記の ACM Communications 誌での特集 [Resnick 97] により、この種のシステムの呼び名として “recommender system” が定着した。また、このころには NetPerceptions や Firefly などの企業によってシステムの商業化も始まった。Web を通じた各種サービスの機能で活用されたり [Lawrence 99, Linden 03, Das 07]、セットトップボックスなどの機器に組み込まれたり [Ali 04] している。2000 年代以降は、物理的な店舗面積に商品数が制限されない電子商取引の発展や、大量の画一的な商品から、少量多品種を扱う mass customization への消費傾向の変化に伴って、その重要性も広く認識されるようになった。このことを象徴する Amazon.com CEO の Jeff Bezos の発言を引用しておこう [Ben Schafer 01]^{*1}。

If I have 3 million customers on the Web,

I should have 3 million stores on the Web

Web に 3 百万人の顧客がいるなら、3 百万の Web ストアを用意すべきだ

現在では、推薦システムは多方面で利用されるようになり、研究も継続的に行われ、多様な方法が目的に応じて考案されている。

この推薦システムには大きく三つの要素技術が関連している。一つ目は、人間から必要な情報を収集し、人間との対話を扱うヒューマン・コンピュータ・インターフェース技術。二つ目は、収集したデータから推薦情報を生成し、それを目的に応じて変換する機械学習、統計的予測、そして情報検索の技術。三つ目は、推薦に必要な情報を蓄積し、処理し、流通させる基盤技術であるデータベース、並列計算、そしてネットワーク関連の技術。本稿では、三つ目の基盤技術については扱わず、推薦システム独自の側面が強い、ヒューマン・コンピュータ・インターフェース技術と機械学習関連技術を中心に主立った研究を紹介する。

^{*1} J. Riedl のメールによれば、J. Bezos は、この発言を幾つかの講演で行った。ここでは、300 万人と書いたが、そのときどきの顧客数に応じて、この数字は変えて用いられた。

第2章

推薦システムの分類と目的

本節では、推薦の個人化の度合いの分類と、運用側と利用者側のそれぞれ目的によるシステムの分類について述べる。これらの目的を考慮して、推薦システムの設計方針を決めることになる。

その前に、推薦システムに関連する技術と、それらとの違いについて述べておく。まず、情報検索における情報フィルタリングがある。これは、逐次的に入力される情報から、利用者の関心や興味を記述した利用者プロフィールに適合するものを選別する技術 [徳永 99] で、推薦システムと良く似ている。だが、主にテキスト文書を扱い、利用者プロフィールも主に索引語で示す点や、必要なものを取り出すより、不要なものを除外することが主目的である点などが異なる。他に、マーケティングの技術とも関連がある。しかし、マーケティングが供給側の視点に立つのに対し、推薦システムは消費側である。推薦システムの推薦が利用者に受け入れられるためには、推薦が実際に客観的なものであり、また、そのことを利用者に示す工夫が必要になる。また、マーケティングでは、レポートなど全体の傾向などを分析して報告することが必要だが、推薦システムにはそうしたことは要求されない。また、マーケティングでは、顧客を客層に分類し、それに応じた対処を行うが、推薦システムでは、利用者のグループ化は目的ではない。

2.1 推薦の個人化の度合い

最初に、推薦の個人化の度合いの3段階 [Ben Schafer 01] を示す。

非個人化 (no personalization)

全ての利用者について、全く同じ推薦をする場合である。例えば、編集者による推薦や、売り上げ順位リストなどである。推薦システムというときは個人化かつ自動化されたものを想定するかもしれないが、こうしたものも広義には含める。非個人化の推薦の例を図 2.1(a) と (b) に示す。図 2.1(a) 中の、(1) は店舗の編集者が手作業で選んだもの、(2) は予約や売り上げの順位リストである。システム側だけでなく、図 2.1(b) のように、他の利用者が手作業で作成する推薦リストもある。

一時的個人化 (ephemeral personalization)

システムを利用する一つのセッションで同じ入力や振る舞いをした利用者には、同じ推薦をする場合である。一時的個人化の推薦の例を図 2.1(c) に示す。この例では、利用者がある本を閲覧するという行動をシステムに対してしたとき、その本に関連する情報を示している。図中の、(1) はこの本と関連が深い本を推薦し、(2) には、書誌情報や売り上げ順位など、(3) では他の利用者の評価やコメントなど、この本についての関連情報を提示している。

永続的個人化 (persistent personalization)

たとえ同じ入力や行動をシステムにしている利用者でも、利用者の個人情報や過去の利用履歴に応じて異なる推薦をする場合である。例えば、過去の購入・利用履歴に基づいて推薦したり、年齢に応じて推薦するアイテムを変えたりする。永続的個人化の推薦の例である図 2.1(d) では、この利用者の過去の商品への評価に基づいて、関心があるであろう本を予測し、順位付けして提示している。



(a) 非個人化 (iTunes Store)



(b) 非個人化 (iTunes Store)



(c) 一時的個人化 (Amazon.co.jp)



(d) 永続的個人化 (Amazon.co.jp)

図 2.1: 個人化の度合いの異なる推薦の例

(a), (b), (d) は 2007/07/26 日に, (c) は 2007/08/04 日にスクリーンショットを取得した。

2.2 推薦システムの運用目的の分類

次に、推薦システムを、運用側の目的に基づいて分類する。文献 [Ben Schafer 01] では、自動・手動の推薦システムを次の5種類に分類し、目的に応じて適切に組み合わせるべきと述べている。

概要推薦 (broad recommendation)

これは、全体の統計情報や運用者側の編集者を使う場合である。全体の統計情報とは「今週の売り上げランキング」や「個々の商品の売り上げ順位」といったもので、編集者による推薦には「評論家が推薦する映画」や「特売品リスト」などがある。図 2.1(a) はこうした概要推薦の例である。こうした概要推薦は、システムを利用し始めたばかりか、ごくまれにしか利用しないような利用者を対象とする。これらの利用者が、自身の要求との関連性を見いだして、システムの利用を続けてもらえるように、大まかな情報を提供するのが目的である。よって、積極的に探さなくても見えるように、システムにアクセスしたとき最初に見える場所に配置すべき推薦である。加えて、利用者が積極的にシステムに働きかけることは期待できないので、非個人化または、アイテムのカテゴリを示す程度の入力に対応した一時的個人化した推薦をすべきである。

利用者評価 (user comments and rating)

これは、運用側が用意したシステム上で、利用者間での推薦をする場合である。例えば、読者の批評文や★の数で示した評価レートをつけられるようにし、それらを他の利用者が参照したり、平均評価値などの統計情報を見たりできるようにする。図 2.1(b) のような他の利用者による推薦リストや、図 2.1(c) の (3) のように他の利用者による評価やコメントなどがこの種の推薦の例である。一般に、運用側のシステムが第三者的な立場で推薦しているとは、利用者にはあまり思われぬ。それに対し、他の利用者の方がずっと信用され、その推薦も受け入れられやすい。運用側には、システムに対する利用者の信頼を高めた

り、利用頻度を高めたりといった利点がある。こうした推薦は、運用側の関与は少なく、評価値順のアイテムリストなどの非個人化か、閲覧中のアイテムの評価値やコメントなどの単純な一時的個人化が行われる。

通知サービス (notification service)

これは、利用者がシステムを操作していないときに、電子メールなどで推薦を配送する場合である。一つには、過去の購買履歴に基づいて、新規のアイテムの中から推薦リストを生成し、利用者に送付するといった、永続的個人化をするものがある。また、利用者が予め設定した条件、例えば、ファンである歌手などを設定しておく、その歌手の新譜の案内が届くという一時的個人化をした推薦もある。これらの推薦は、システムの再利用を利用者に促すことを目的とする。

関連アイテム推薦 (item-associated recommendation)

利用者が注目している、例えば、電子商取引サイトで、商品を閲覧しているとか、「買い物かご」にすでに商品を入れている状況を想定する。このとき、注目しているアイテムの比較候補（例：他メーカーの同等品）を示して、購入の判断を助けたり、購入の決断を促す場合や、補足的な商品などを提示して cross-selling（例：ハンバーガーにポテトなどの関連品を薦めて同時に購入させる）を促す場合がこの種の推薦である。例としては、図 2.1(c) の (1) のように閲覧中の商品の関連商品を示すものがある。注目している商品の情報だけに基づく一時的個人化も、利用者の過去の行動も考慮する永続的個人化も可能である。

緊密な個人化 (deep personalization)

システムが積極的に利用者の情報を能動的に収集したり、過去の行動の情報を蓄積し、それらに基づいて推薦を行う場合である。図 2.1(d) のような、個人向けの推薦リストなどが、この種の推薦に該当する。利用者がシステムを利用し続けることで蓄積された情報に基づいて、永続的な個人化が可能になる。すると、より適切な推薦ができるようになり、他のシステムとの差別化につながり、利用者の長期間にわたるロイヤリティ構築に役立つ。だが、一方で、最も実装

に困難が伴い、コストも要する。

2.3 推薦システムの予測タスクの分類

推薦システムが達成するタスクに基づく分類を文献 [Herlocker 04, Gunawardana 09] に沿って示す。

適合アイテム発見 (finding some good items)

このタスクの目的は、利用者が自分の嗜好に適合するものを、何か見つけ出すことである。利用者が積極的な動機を持って、情報を見つけるために推薦システムを利用するときは、こうした目的になる。

例えば、今から食べに行く店を決めるために、レストラン推薦システムを利用することが想定される。この場合には、予測される評価の高い、比較的少数のものに絞り込んで利用者に提示する。利用者はこのリストを上位から閲覧することで、自身の決定に必要な情報を知ることができるだろう。

評価値予測 (predicting ratings)

このタスクの目的は利用者がアイテムに付けるであろう評価値を予測することである。何らかのフォーマットに従ってアイテムが整列されていたり、構造化されている状況を考える。例えば、新商品を発売日が直近になるように整列したり、カテゴリ別に商品を分類したり、メールや掲示板で記事をスレッド状に表示したりする場合である。このように整列されたアイテムを、利用者が閲覧しているとする。このとき、閲覧の順序を決めたりとか、どの記事を閲覧するかとかを決めるために、手がかりとなる情報があれば便利である。そのような情報として、利用者が関心をもつ度合いを、システムは予測して付随的に提示する。適合アイテム推薦とは違って、利用者には、必ずしも、最終的に意志決定をするような積極的な動機があるとは限らず、また、関心のあるものを特に選んで表示するといった、アイテムの提示フォーマットを変えるような要求も

ない。

例えば、決まった予定はないが、レストランの紹介 Web サイトなどを閲覧しているときなどである。レストランは、料理の種別や、推薦の度合いは、★の数や、アイコン、グラフなどを対象と共に表示することで利用者に提示する。この推薦情報は、多数のレストランの中から、利用者にとって関心のあるものを絞り込むのに役立つ。

適合アイテム列挙 (finding all good items)

このタスクの目的は、利用者が自分の嗜好に適合するものを網羅的に見つけ出すことである。これは裏返しに、適合しないものを排除する目的であるともいえる。例えば、会社の法務部門が関連する特許や判例を検索したり、スパムメールの可能性がないメールだけを閲覧したいといった場合である。

効用最適化 (optimizing utility)

このタスクの目的は、何らかの効用関数を設定し、それを最適化するようなアイテムを見つけることである。適合するアイテムなら 1、それ以外は 0 という効用関数を考えると、適合アイテム発見はこの効用最適化とみなせるので、適合アイテム発見を一般化したタスクと考えることができる。例えば、電子商取引サイトで推薦システムによって利益を増やす場合に、元から購入を意図していたアイテムに追加のアイテムを購入させるという組み合わせ販売 (cross-selling) を促進するような効用関数を設定したりする。

2.4 推薦システムの利用動機の種類

文献 [Swearingen 01] では、利用者の推薦システムの利用動機の種類を示している。前者のタイプほど、既知のアイテムに近い推薦することが望まれる。

備忘録 (reminder) 既知のアイテムを思い出させる。

類似品 (more like this) 比較などのため既知のアイテムに類似したものを探す。

新規アイテム (new items) 自分が確実に好むであろう，未知の新製品を探す．

視野を広げる (broden my horizon) 他のジャンルにも自分の関心を広げる．

第3章

推薦システム設計の要素

多種多様な推薦システムのためのアルゴリズムが存在するが、一体、どのアルゴリズムを使えばよいのであろうか？ 機械学習の基本的な定理であるノーフリーランチ定理によれば万能アルゴリズムは存在しない。よって、アルゴリズムは、推薦システムを利用する目的や、推薦を実行する環境の制約に応じて選択する必要がある。ここでは、そのために考慮すべき要因を大きく二つに分けて述べる。一つは、推薦の性質である。2.3節の利用者の目的などに応じて、推薦は適切な性質を備えるべきである。この性質を測るための規準を幾つか示す。もう一つは、推薦を計算するためのデータや計算機資源の制約である。データや計算機資源は無限にはなく、何らかのトレードオフを考慮しつつ推薦システムは設計する必要がある。

3.1 推薦の性質

推薦システムでは、利用者が好むものを予測して提示する。だが、利用者が何を好むかは、利用者の目的、システムを利用する状況、推薦の候補などによって変化する。よって、提示する推薦の性質を決めるにあたって考慮すべき規準があれば役立つ。これらの規準を以下に示す。なお、推薦の質の評価については、文献 [Herlocker 04, Gunawardana 09] が詳しい。

3.1.1 予測精度

予測精度とは、予測して推薦したアイテムに、実際にどれくらい利用者が関心をもつかという規準である。利用者が関心のないアイテムを推薦しても役に立たないので、予測精度は最も重視すべき規準である [Swearingen 01]。評価はオンラインで行う場合と、オフラインで行う場合がある。オンラインでの評価とは、被験者に実際にシステムを利用させ、推薦が適合したかどうかを調査するものである。一方、事前に被験者から集めた嗜好データと、予測した結果の一致を調べるのがオフラインでの評価である。前者の方がより実際の運用に近い評価ができるが、この調査のコストは高い。そのため、得られるデータ数が少なく安定した検証が困難であったり、多数の項目での比較が困難になるなどの問題がある。こうした点ではオフラインでの評価の方が有利になる。

オンラインの評価では、それぞれのアルゴリズムを利用して、推薦を利用者が受け入れる比率で、アルゴリズムの相対的な予測精度を評価する。また、二つのアルゴリズムによる推薦リストの上位から交互にアイテムを選んで、一つの推薦リストにまとめる。そして、利用者にそれを提示して、どちらのアルゴリズム由来のアイテムがより頻繁に選ばれるかといったことで評価する方法などがある。

オフラインの評価では、一般の機械学習と同様に、交差確認によって汎化誤差を推定し、その汎化誤差で予測精度を評価する。なお、予測精度の評価については [元田 06] の5章に詳しい。ここで、アルゴリズムに超パラメータがある場合の注意を述べておく。超パラメータとは、9.1節の方法の近傍の大きさといった、アルゴリズムで調整すべき変数のことをいう。通常の交差確認では、データを訓練用とテスト用の二つに分けるが、超パラメータがある場合は、厳密には、データを訓練用 (training)、確認用 (validation)、およびテスト用 (test) の三つに分ける必要がある [Hastie 09, 7.2 節][Bishop 06, Bishop 08, 1.1 節]。そして、アルゴリズムの学習には訓練用データを、超パラメータの決定は確認用データを、そして最終的な予測精度はテスト用データを

用いる。特に、新たに超パラメータを導入したが、予測精度の向上がわずかな場合には、こうした厳密な評価実験をしておくことを薦める。

推薦システムでは、次のような尺度が、テスト用データに対する予測精度の評価に利用されている。

正解率 (accuracy)

利用者の関心への適合や不適合が、予測結果とテスト用データで一致した割合を示す。評価値を予測するシステムでは5段階のうち上位2段階のいずれかなら適合とみなしたりする。最も基本的な評価指標である。

精度 (precision) と再現率 (recall)

適合判定されたアイテムのうち実際に適合しているものの割合が精度、全ての適合アイテムのうち適合と判定されたものの割合が再現率である [徳永 99]。適合アイテムを一つ見つければ良い適合アイテム発見タスクでは精度を、全ての適合アイテムを見つけない適合アイテム列挙タスクでは再現率を重視すべきである。情報検索で利用されている、精度や再現率に基づいた、F 尺度 (F measure) や ROC 曲線 (receiver operating characteristic curve) なども用いられている。

平均絶対誤差 (Mean Absolute Error)

テスト用データの評価値と予測した評価値の差の絶対値のテスト用データ上での平均である [Herlocker 04]。評価閲覧タスクでは、適合か不適合かの判定より、評価値そのものを利用者は見る。よって、このタスクでは、評価値の予測精度のずれを評価するこの指標を重視すべきである。

half-life utility metric と順位相関 (rank correlation)

これらは、推薦するアイテムの並び方の良さを評価する指標である。計算方法の詳細は文献 [Herlocker 04] を参考にされたい。適合アイテム発見を目的とする場合、推薦システムには意志決定支援の側面が要求される。このときは、評価値自体よりも、評価の大小を重視すべきである。これらの指標はこうした場

面で有用である。

最後に、推薦システムの予測精度の評価についての問題を指摘しておこう。交差確認による評価では、テストに使ったサンプルと、今後予測するサンプルは同じ分布から得られることを、通常は仮定している。詳細は5.2節で述べるが、評価されていないアイテムは、利用者が関心を示さなかったものであることが多い。そのため、評価値が欠損するアイテムは、評価が低いアイテムに偏る傾向がある。こうした原因により、実際にシステムが稼働して予測対象となるアイテムの分布とテストに用いられるアイテムの分布は異なり、厳密に予測精度を評価することは難しい。そのため、わずかに予測精度を向上させる試みは実用的には利益がないことが多い。1%なり3%ほどの危険率で統計的に有意な差がなければ、ほぼ同等の予測精度とみなし、他の規準を重視して推薦システムを設計すべきである。

3.1.2 多様性・セレンディピティ

利用者が関心を持つであろうアイテムを推薦することは推薦システムの目的であり、上記の予測精度はこのことを定量的に評価する。それに加えて、多くの場合、利用者が知っているアイテムを推薦してもあまり有用ではない。よって、関心があることに加えて、推薦には、**目新しさ (novelty)**、すなわち、わかりきったものではないことが要求される。例えば、利用者がスピルバーグ監督のファンであり、この利用者にスピルバーグ監督の新作映画を推薦したとする。このとき、利用者はこの映画に関心を持ち、まだ知らない目新しい推薦であり、上記の二つ条件を満たしている。さらに、要求される条件にセレンディピティの高さがある。推薦における**セレンディピティ (serendipity)**とは、この目新しさに、思いがけなさ、予見のできなさ、または意外性の要素が加わった概念である。例えば、スピルバーグ監督とよく似た作風の新人監督の作品を考える。このとき、作風が似ているため利用者はこの作品に関心を持ち、また新規性もある。さらに、利用者はこの新人監督の作風がスピルバーグ監督と似ていることを知らないため、この作品が推薦されることを予見できない、すなわち、意外

性がある。よって、この推薦にはセレンディピティがあるといえる。

しかし、このセレンディピティに伴う感情的な応答を定量的に評価することは難しい。それでも、セレンディピティの一側面ではあるが、それらを定量化する試みもある。それらの中でも、**多様性 (diversity)** [Hurley 11] は、文献 [Ziegler 05] による提案以降多くの研究が行われている。多様性は、推薦リストに含まれるアイテムが互いに似ていないことである。あるアイテム間の類似度を決めて、推薦リスト内のアイテム間の類似度を集約することで、多様性は定量的に評価することが多い。その他、グループ向けの推薦には現れないが個人向けの推薦には現れるもの [Herlocker 04] や、単純な予測器では候補にならないが、高度な予測器では候補になるもの [村上 07] などセレンディピティが高いとみなし、こうした仮定に基づいた定量的評価尺度を提案している。

一般に、利用者が推薦を採用したとき、その結果不満だったときのコストは低いが、満足したときの利得は大きい分野では、セレンディピティを重視すべきである。映画や音楽など娯楽に関する推薦では、こうした状況になることが多い。

3.1.3 被覆率

被覆率 (coverage) とは、全アイテムのうち、評価値の予測が可能なアイテムの割合である。評価値の予測が不可能な状況には、以下のようなものがある。協調フィルタリング (8章) では、他の利用者の評価値を利用する。すると、誰にも評価されていないアイテムは評価の対象にできない。もう一方の内容ベースフィルタリング (12章) では、アイテムの特徴量が欠損していたり、利用者のプロフィールが未整備である場合には、評価値を予測できない。適合アイテム発見タスクでは、利用者が満足するものが何か見つければ良いので被覆率は比較的低くても問題は生じない。評価閲覧が目的なら、評価値のないアイテムが多数あるのは不便なので被覆率は高くあるべきである。適合アイテム列挙タスクでは、推薦すべき対象の見落としは許されないため、基本的に被覆率は 100% でなければならない。

3.1.4 学習率

嗜好データの増加に伴って予測精度は向上するが、その向上の度合いを**学習率** (learning rate) と呼ぶ。これは、実用的な予測精度に達するまでに必要な嗜好データの数で決まる。システム全体を評価する学習率は、嗜好データの総数から計算するが、特定のアイテムや利用者に限定した評価をするための学習率も用いられる。学習率を調整するパラメータをもつアルゴリズムも多いが、学習率を高くしすぎると過学習のため汎化誤差が悪化して、予測精度の向上が不十分なレベルで止まる場合もある。しかし、利用者が評価付けをあまりしない場合や、服飾品など商品のサイクルの早いアイテムでは、過学習の危険性があっても学習率はやや高くすべきである。

3.1.5 推薦の性質に関するトレードオフ

言うまでもなく、正解率やセレンディピティなど上記の規準で全て良いものが理想的な推薦システムである。しかし、これらの評価規準は、次に挙げるようなトレードオフの関係にあり、目的に応じてバランスをとる必要がある。予測精度は、推薦において最も重要な規準だが、これだけでは不十分であることは十分に注意すべきである。文献 [Cosley 03] には、利用者は5段階評価で1段階良く、もしくは悪く改竄した推薦を見せられると、そのことに利用者は気づき、このような改竄システムへの利用者の満足は低いことが報告されている。よって、予測精度は、利用者の満足に影響していることは確かである。だが、計測した予測精度が同じシステムでも、利用者の満足には大きなばらつきがあるとも、この文献は報告している。ほとんどの利用者が好むであろう限られたものだけを推薦すれば、予測精度は一般に高くなる。例えば、スーパーマーケットでの買い物で牛乳や卵など、ほとんどの顧客が購入する商品を推薦すると、予測精度の観点からは良い推薦である。だが、当たり前すぎて目新しさはなく、一概に良い推薦とはいえない。新たにシステムを利用し始めた利用者には、システムへの信頼を高めるために、予測精度を重視して確実に好まれるものを推薦する方が良

い。予測精度の評価指標は、評価値を計算できなかったアイテムは無視して計算するのが一般的である。そのため、嗜好データが十分なアイテムだけを推薦対象にすれば、予測精度は向上するが、被覆率は下がってしまう。また、学習率の向上も過学習などの影響で、予測精度を低下させる場合がある。以上のように複雑なトレードオフの関係があるため、どの指標を、どのようなバランスで重視するかは、推薦対象や、利用者の目的など多くの要因を考慮して決めなければ、利用者の満足を得られるような推薦システムは設計できない [McNee 06a].

こうした設計を組織的に行うための研究もいくつかある。文献 [McNee 06b] では、Human-Recommender Interaction (HRI) というモデルを提案している。これは、システムと人間のやりとりである推薦ダイアログ (recommendation dialogue)、推薦の傾向を表す推薦器の個性 (recommender personality)、および利用者情報探索タスク (user information seeking task) の三つ点について、それぞれの特徴を記述するための規準を定めている。この規準に基づき、利用者の推薦への要求や、推薦アルゴリズムの特徴を記述し、目的に応じて適切な対応付けをする HRI 解析プロセスモデルを提唱している。

3.2 推薦候補の予測に関する制約

推薦候補を予測するために必要な、データや計算機資源は無限にはなく、何らかのトレードオフを考慮しつつ推薦アルゴリズムを選択する必要がある。よって、これらの制約や条件についてまとめる。

3.2.1 嗜好データの制約

嗜好データの最も顕著な特徴は非常に疎 (sparse) であることである。すなわち、非常に多くのアイテムが存在するが、利用者が評価しているのはごく一部で、その他のアイテムへの評価値は欠損している。具体的には、評価値があるのは全体の 1%~0.001% のオーダーである [Ben Schafer 01]。また、欠損は均一ではなく、Zipf の法則

[徳永 99]のように、被評価数の順に、被評価数ごとのアイテム数を整列すると、被評価アイテム数は指数的に減少する現象がみられる [Weigend 03]。こうした疎なデータからの予測は困難である。また、詳しくは5.2節で述べるが、嗜好の評価値は、統制されていない環境で採取された心理的な量なので、揺らぎが大きく、評価のたびに変化して不整合を生じる問題もある。あと、利用者数とアイテム数の比率は予測精度に影響する [Herlocker 04] ので、実際の運用状況に合わせてテストをすべきである。最後に嗜好データの更新の問題がある。推薦システムは運用中に、随時嗜好データが追加される。また、新たに利用者やアイテムがデータベースに追加されることもある。こうした変化に応じて予測モデルを更新する必要がある。平滑化などを用いた予測技術を使うと、疎なデータでも比較的安定的な予測ができるが、計算量が増えて予測モデルの更新を頻繁に実行できず、これらの変化に対応できなくなるといった問題もある。

3.2.2 その他の制約や条件

データ数が多数であるにもかかわらず、高速な予測が要求されるスケーラビリティは重要な問題である。利用者数は10万~100万、アイテム数は10~100万、利用者あたりの評価数10~1000という大規模なデータにもかかわらず、10~1000の要求に対して、10~100ミリ秒の時間で応答することが要求される [Ben Schafer 01, Linden 03]。このような高いスケーラビリティを達成しつつ、正確に予測することも困難な課題である。

他に、推薦を利用する状況の問題もある。例えば、レストランの推薦システムでは、一人で食べに行く場合と、家族で食べに行く場合は異なった推薦をすべきだろう。こうした推薦をする状況や利用者の暗黙的な要求を考慮するかどうかは大きな要因となる。また、利用者がどれくらい詳細な推薦を求めているかも、アルゴリズムを選択するときに考慮すべき事柄である。すなわち、利用者の嗜好に適合か不適合の2段階程度の大まかなものでよいのか、購入のための意志決定のため、いろいろな評価項目について利用者の嗜好への適合度を詳細に要求するのかといった違いである。

第 II 部

推薦システムの実行過程

第 4 章

推薦システムの実行過程

ここまでは、推薦システムを分類し、設計にあたって考慮すべき事項について述べてきた。ここでは、推薦システムがどのように実現されるかを見てゆこう。推薦システムは、図 4.1 のように、データの入力、嗜好の予測、そして推薦の提示の三つの段階で推薦を行う。これは **O-I-P モデル** (output-input-process model) [Konstan 03] とも呼ばれる。以下、これらの各段階の概要について述べる。

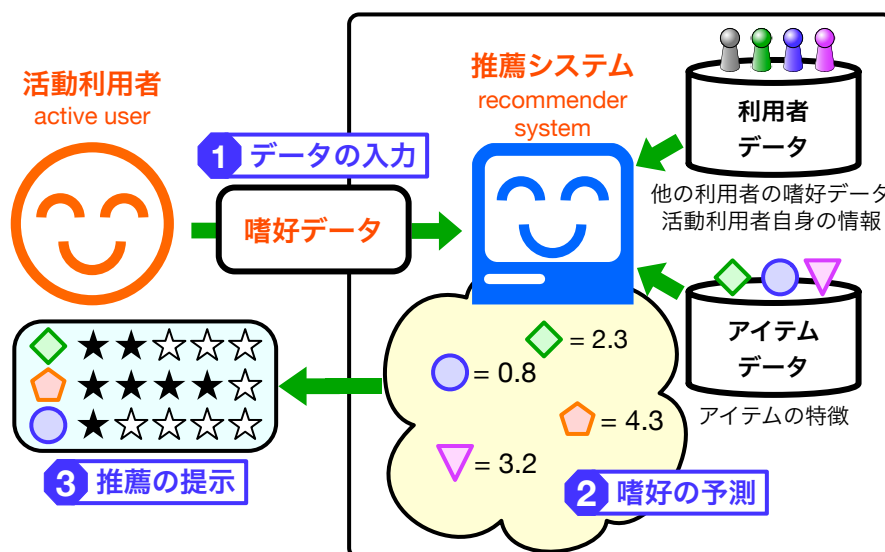


図 4.1: 推薦システムの実行過程 (O-I-P モデル)

データの入力

推薦システムを利用して、推薦を受けようとしている人を**活動利用者** (active user) と呼ぶ。活動利用者は自身の**嗜好データ** (preference data) を推薦システムに入力する。嗜好データとは、いろいろなアイテムについての関心や好みの度合いを数値化したデータである。この嗜好データの代わりに、関心のあるアイテムについてのより具体的な記述を検索質問や批評として、活動利用者に入力させるシステムもある。また、推薦には、他の利用者の嗜好データ、アイテムの特徴データ、活動利用者自身の情報、推薦の状況や目的なども利用される場合があり、これらの情報を収集する場合もある。この段階については5章で述べる。

嗜好の予測

活動利用者の嗜好データに加え、収集しておいた利用者の他の情報やアイテムの情報を利用して、活動利用者がまだ知らないアイテムへの、活動利用者の嗜好を予測する。嗜好の度合いを数値として予測する手法や、単に好きか嫌いかの識別をするだけの手法がある。実現手段としては、機械学習の手法を用いるものや、人手によるルールを用いるものがある。この段階については概要を6章で述べ、各種のアルゴリズムを第III部で紹介する。

推薦の提示

予測した嗜好に基づいて、目的に応じた適切な形式で、推薦結果を活動利用者提示する。このために、2.2節や2.3節のいろいろな目的に応じた表示形式の変更や、3.1節の各種指標のバランスを調整するためのアイテムの選別や順位付けの変更などを行う。この段階については7章で述べる。

第 5 章

データの入力

ここでは、推薦システムの実行過程の最初の段階である「データの入力」について述べる。この段階では、活動利用者に自身の嗜好データを、推薦システムへ入力させる。嗜好データとは、利用者の各アイテムへの関心や好みの度合いをを数量化したものである。システムによっては、この嗜好データの代わりに、活動利用者に検索質問や批評を入力させるものもある。この検索質問は、アイテムの特徴についての制約条件を具体的に記述したものである。例えば、レストランの推薦システムで、価格帯や、和洋中の別などを具体的に指示するために「価格は 6000 円以下で、和食の店」といった形式の検索質問を入力する。こうした検索質問は、情報検索やデータベースのクエリ検索の技術がほぼ転用できるので、ここでは、嗜好データについて述べる。さらに、これら嗜好データや検索質問で表された、活動利用者の嗜好パターンの以外のデータも推薦システムは利用する。このようなデータとして、活動利用者以外の利用者の嗜好データ、アイテムの特徴、利用者の年齢や性別などの情報、現在位置などの利用状況を示す情報などがあり、これらについても述べる。なお、嗜好データの収集全般については、文献 [土方 04] にまとめられている。また、文献 [Swearingen 01] は、推薦システムの利用者へのアンケート調査結果に基づいて、入力インタフェースについての設計指針を示している。アイテムについての情報は、利用者が評価するときに見えるようにしておくこと、システムへの満足が高まると報告している。

表 5.1: 嗜好データ獲得法の長所と短所

	明示的	暗黙的
データ量	×:少ない	○:多い
データの正確さ	○:正確	×:不正確
未評価と不支持の区別	○:明確	×:不明確
利用者の認知	○:認知	×:不認知

5.1 暗黙的と明示的な嗜好データの獲得

まず、嗜好データを獲得するアプローチは、おおきく暗黙的と明示的の二種類に分けられる。明示的な獲得とは、利用者に好き嫌いや、関心のあるなしを質問し、利用者に回答してもらう方法である。もう一方の暗黙的な獲得とは、利用者の行動をから、利用者の嗜好や関心を推察することで嗜好データを得る方法である。例えば、購入したり、閲覧したりしたアイテムには、利用者は関心があるとみなしたりする。

まず、二つの嗜好データの獲得法を比較する。これらの獲得法の長所と短所を表 5.1 にまとめた。データ量については、利用者の嗜好の予測には統計的な方法が用いられるので、予測を正確にするにはより多くのデータを収集できた方が有利となる。しかし、質問に答えるといった手間を利用者は嫌うことが多いため、明示的な獲得では多数のデータの収集は難しい。よって、これらの点では暗黙的な手法が有利である。

データの正確さについては、暗黙的な獲得では、誤ってクリックしてしまったとか、人に頼まれて購入したなどの理由で、本当は関心がないものも、関心があるとみなされてしまう場合がある。このため、収集されたデータの正確さにおいては明示的な獲得が優れている。

利用者に明示的に評価してもらう場合では、アイテムを利用者が評価したかどうかはもちろん明確である。しかし、暗黙的な評価では、利用者がそのアイテムに対して積極的な行動をしなかったことをもって、そのアイテムへの不支持とみなす場合がある。

例えば、閲覧しなかったアイテムは好きではないとみなしたとする。このとき、アイテムについて未評価であることと不支持であることの区別ができない。場合によっては、閲覧していないために、利用者が好むアイテムを嫌いだとシステムがみなすこともある。

最後の利用者の認知とは、利用者が自分の嗜好データをいつ、どのように取得されたかを知っているかどうかということである。システムが提示した推薦は、利用者がその根拠を把握していた方が受け入れられやすい。暗黙的な獲得では、嗜好データを意識的に入力していないので、推薦が根拠なくなされたもののように感じられやすく不利である。

5.2 明示的な獲得

アイテムを利用者に提示し、利用者にそのアイテムに対する好みの度合いを答えてもらう明示的な嗜好データの獲得について詳細を述べる。

評価の動機付け

明示的な獲得法では、利用者はアイテムを評価することを面倒だと思うので、暗黙的な方法に比べて多数の嗜好データを集めにくいと述べた。[Swearingen 01]では、利用者は推薦の精度が向上するなど、評価付けによるメリットが明確であれば、ある程度の手間をかけて評価付けをするとの調査結果を報告している。よって、利用者に評価をさせるような動機付けは重要である。

自身への推薦の精度を向上させるということは、利用者にとって主な動機付けとなる。だが、管理者が想定するようなこの動機以外にも、自分の意見の表明をするためや、他の利用者の手助けになるということを動機とする場合もある [Herlocker 04]。これらの動機は、利用者の評価数の順位の公開などによって喚起することができる。さらに、明示的にインセンティブを与えることも考えられる。文献 [Melamed 07] は、情報検索の結果の順位付けに、他の利用者の評価を利用する、一時的個人化の推薦シ

システムを提案している。このシステムでは、市場の考えが導入され、検索結果を閲覧するには、ポイントの支払いが必要である。閲覧する文書の、被評価数が多く、評価が高いほど多くのポイントを支払う必要がある。一方、ポイントは、検索した結果を評価することで獲得でき、被評価数が少なく、高い評価をすると獲得ポイント数は増える。すると、検索をするために、検索結果の評価をする必要が生じるため、積極的な評価付けが期待できる。

5.2.1 採点法と格付け法

利用者が好みの度合いを答えるには、それを測る尺度が必要になる。好みの度合いを表す尺度として、0～5や-3～+3のような数値尺度を使う**採点法** (scoring method) や、上・中・下や適合・不適合などの順序付カテゴリ尺度を使う**格付け法** (rating method) [竹内 89] が良く利用されている。こうした方法は人間の聴覚や味覚などを定量的に計測する官能検査 (sensory test) の分野で研究されてきた [佐藤 85]。採点法や格付け法は、単純な入力フォームを用いて、比較的多数のアイテムに対する嗜好データを得られることが利点である。

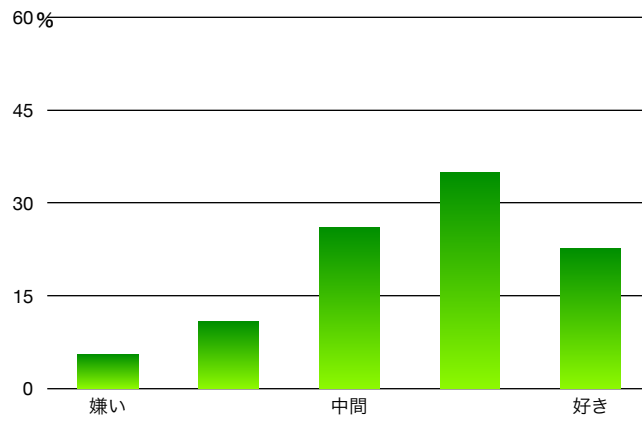
これらの方法を使ううえでの注意点を幾つか述べておく。文献 [Cosley 03] では、利用者は評価尺度の目盛りが細かい方を好む傾向があること、さらに、細かい評価で予測精度が向上することはないが悪くもならないことを報告している。よって、目盛りは細かめに設定することを推奨している。さらに、-3～+3の尺度で、中立の0を抜いた尺度を使うと、中立の評価の多くは弱い肯定的な評価+1に移されること、予測評価値を見せながら評価させると、利用者はそれに「引きずられた」評価をすることも報告している。2.3節で述べた適合アイテム発見を目的とする場合、目的に適合/不適合の2段階でも十分な場合が多いが、評価閲覧タスクでは、どれくらい不要なアイテムを除外したいかは利用者次第なので、より詳細な多段階の尺度を用いる方が良いだろう。次に、質問の仕方にもいろいろな配慮をすべきである。例えば、採点法では等間隔の尺度を連想させるように、等間隔の目盛りを見せるなどの工夫がある。これ

らの配慮については中森の [中森 00] を参考にされたい。

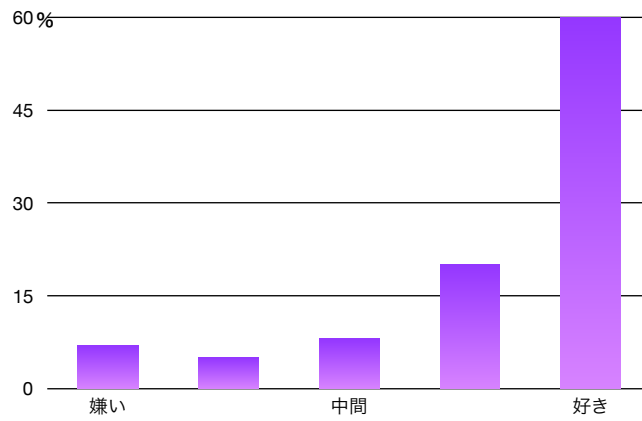
5.2.2 評価値の揺らぎや偏り

採点法や格付け法は大量の嗜好データを比較的容易に得られるので多用されてきたが、当然ながら欠点もある。先に、明示的な獲得は暗黙的な獲得と比べてより正確に利用者の嗜好を評価できると述べた。だが、絶対的には不正確さや揺らぎがある。真の嗜好の度合いは、脳の活動を直接観測するなどすれば将来的には計測できるであろうが、現在のところは厳密には計測できない。そのため、揺らぎがあるかどうかの直接的な証明はできない。よってここでは、採点法や格付け法によって計測した評価値が、真の評価値と乖離している間接的な証拠と、その乖離の原因を示す。

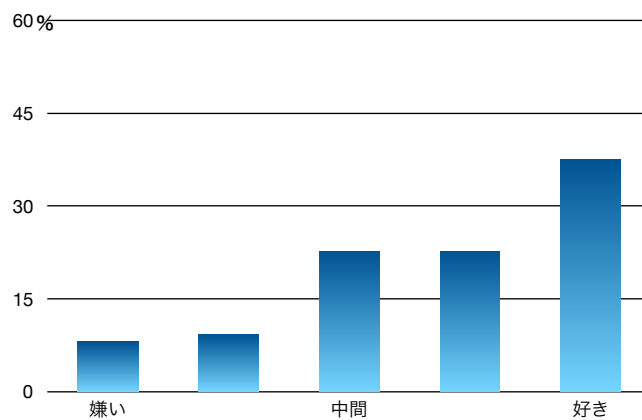
まず、評価値の揺らぎの証拠を示す。官能検査の研究では、たとえ被験者が同じ評価値を与えていても、人によって嗜好の強さが違っていたりとか、時間がたつと一貫性が保たれなくなる問題があることが経験的に知られていた [Luaces 04]。ソムリエなど訓練された被験者が、同一セッション内、すなわち、時間をあけずに連続して評価した場合でなければ、尺度を一定に保つことは難しいとされている。嗜好データについても、一度映画を評価付けしたあと、6週間後にもう一度同じ被験者に同じ評価付けさせると、二つの評価値の間の相関は 0.83 であったとの報告がある [Hill 95]。文献 [Cosley 03] でも同様の報告がされている。同一セッション内でも、寿司の嗜好について採点法で尋ねたのち、無関係な質問を幾つかしてから、下記の順位法で再び同じアイテムについて嗜好を質問すると、68.3% の被験者の回答に不一致が観測された [神島 04]。これらの実験結果は、嗜好データには揺らぎがあることを示している。他に、代表的な映画評価データにおいて、いろいろな工夫をしても、平均絶対誤差 (MAE) を 5 段階尺度で 0.73 の「魔法の壁 (magic barrier)」より小さくできないことから、評価値そのものに揺らぎがあることが示唆されている [Herlocker 04]。以上のように、絶対的な評価値を使う採点法や格付け法では、被験者は、質問時期の違いなどにより揺らぎが生じるといえるだろう。



(a) MovieLens [MovieLens]



(b) Amazon.com [Weigend 03]



(c) 寿司 [Kamishima]

図 5.1: アイテムへの評価値の分布

次に、評価値の偏りについて述べる。図 5.1 に、5 段階の採点法を用いた 3 種類の嗜好データの、評価値の分布を示す。それぞれ、(a) MovieLens の 100 万要素のデータ集合 [MovieLens], (b) 電子商取引サイト Amazon.com [Weigend 03], (c) 寿司の嗜好調査 [Kamishima, Kamishima 06] での分布である。どのデータでも、『好き』の方へ明らかに偏っている。この偏りの原因には、サンプリングと真の嗜好からの乖離の二つが考えられる。サンプリングの偏りの原因として、図 5.1(a) や (b) では、利用者が、関心がある選択的にアイテムを評価していることや、図 5.1(b) や (c) では、市場の淘汰を受けて人気のあるアイテムのみが評価候補となっていることが挙げられる。このようなサンプリングの偏りは、3.1.1 節で指摘したように、予測誤差の正確な評価を妨げる。真の嗜好から乖離する理由としては、利用者個人がもつ心理効果の影響が考えられる。例えば、過剰な酷評は社会的通念的に良くないとの考えをもつ人には、全体的に評価が高くする寛大効果 (leniency effect) が見られ、あいまいな判断や質問では中心のスコアを選びやすい中心化傾向 (central tendency) などが生じる [中島 99]。さらに、質問の仕方による影響も考えられる。例えば、尺度の一方だけが連続して選ばれるように質問を配列すると偏りが生じる場合がある [中森 00]。しかし、推薦システムでは、推薦結果の提示と嗜好データの収集を兼ねるため、利用者が好むと予測される順にアイテムを並べ評価付けさせることがよく行われる。すると、高い評価値が高頻度で連続してしまう。このように、設計上の制限により、偏りを生じるような質問の仕方をしてしまうという問題もある。

5.2.3 順序の利用

そこで、採点法や格付け法以外の調査方法の利用が考えられる。文献 [Herlocker 99] では、利用者の類似度評価においてスコアの順位関係だけを考慮することや、利用者ごとの平均評価値を 0 に正規化することで予測精度が向上することを報告している。このことは、採点法で得た評価値の絶対的な値ではなく、相対的な大小が重要であることを示唆しているといえるだろう。また、採点法や格付け法で得られる量は、

本質的には大小関係にのみ意味がある順序尺度 [Stevens 51, 鷺尾 98] であると指摘されている [中森 00]. そこで、好きなものから嫌いなものへ順に、複数の対象を並べるという**順位法** (ranking method) を利用する「なんとなく協調フィルタリング」 [Kamishima 03b, Kamishima 06] を神嶋は提案した. 少なくとも調査したデータにおいて、順位法の採用で予測精度が向上した. ただし、順位法にも問題点がある. 同時に多数のアイテムを整列するのは難しいので、大量の嗜好データをまとめて得ることは難しい. また、評価は常に相対的で、絶対的な評価は得られない. そのため、相対的に良いものを選ぶような意志決定には役立つが、絶対的な評価が求められる評価閲覧タスクなどには向かない.

官能検査の調査方法としては、取り出したアイテム対のどちらが良いかを指定する一対比較法 (paired comparison) や、幾つかの候補の中から最良のものを指定させる択一法 (choice method) などもある. これらの方法についての研究は著者はまだ知らず、今後の研究が待たれる.

5.3 暗黙的な獲得

暗黙的な嗜好データの獲得では、アイテムに関連した、利用者の行動に基づいて、そのアイテムについての嗜好を判断する. 利用者があるアイテムを閲覧したり、購入したりすると、これらの行動はそのアイテムへの潜在的な肯定を示していると考えられる. また、購入は閲覧よりもより強い肯定を示すとも考えられる. このような行動による潜在的な嗜好の強弱を Nichols は論じている [Nichols 97, Burke 02]. 強い嗜好を表すものから順に次のような行動を挙げている.

- | | | | |
|---------------------|------------------|-----------------|---------------|
| 1. Purchase | 2. Assess | 3. Repeated Use | 4. Save/Print |
| 5. Delete | 6. Refer | 7. Reply | 8. Mark |
| 9. Terminate Search | 10. Examine/Read | 11. Consider | 12. Glimpse |
| 13. Associate | 14. Query | | |

文献 [Cho 07] では、閲覧、ビデオのプレビュー、購入の3種類の行動それぞれを暗黙

的な肯定入力と考え、それぞれの行動別に利用者間の類似性を計算している。

他の暗黙的な獲得法として次のようなものがある。推薦リストの上位から A, B, C, …とアイテムを閲覧して C を選んだとき、A や B を見たにも関わらず C を選択したことから、A より C, B より C を好むという相対的嗜好順序を得る方法 [Joachims 02] が提案されている。閲覧するという行為だけでなく、その時間を計測することで、より詳細な情報を得る方法などもある。また、新たに入力装置を導入して、利用者の行動情報を収集し、そこから暗黙的に嗜好データを獲得する試みもある。例えば、マイクで収集した発話内容 [高間 07] や、アイカメラを使って求めた注視領域 [吉高 07] などを利用する試みなどである。

5.4 嗜好データのその他の要因

嗜好データで推薦に影響するその他の要因を挙げておく。利用者が初めてシステムを利用するときに、特定のアイテム群について明示的に質問して、嗜好データを集めることが考えられる。全ての利用者が共通に評価しているアイテム群があると、8章の協調フィルタリングでは利用者間の嗜好の類似性を評価しやすくなる利点がある。しかし、音楽のようにその場で少し聞かせて評価できるようなものならよいが、映画などは見たことがないものは評価しにくい。よって、こうした共通アイテム集合を利用できるかどうかは推薦するアイテムの種類に依存する。

評価したときの時間情報も利用できる。服飾品のように流行の影響がある場合には、時間がたった嗜好データはあまり有効ではないだろう。また時間の前後関係に依存する 10.4 節のような嗜好の予測方法もある。単純な好き・嫌いではなく、Zagat のレストランガイドのように、味・サービス・内装といった項目ごとに分けて嗜好データを収集することもできる。こうしたシステムとしては [Adomavicius 07] がある。

5.5 嗜好データ以外のデータ

嗜好データ以外の、推薦に利用されるデータについてまとめておく。

5.5.1 アイテムの特徴

アイテムを特徴ベクトルで記述したデータで、内容ベースフィルタリングでは必須である。推薦対象がテキストである場合は Bag-of-words モデルで tf-idf 重み [徳永 99] が一般に使われる。この場合多数の特徴量が得られるが、個々の特徴の推薦への寄与は小さい。一方、推薦システムの設計者が意図的に選択した特徴は、多数は得られないが、個々の特徴はより推薦に寄与する。こうした特徴には、映画の場合では監督や制作年、ラーメンであればスープや麺の太さといったものが挙げられる。このように、明確な特徴の他に、アンケート調査などで獲得した印象などを特徴にする場合もある。映画の例では「悲しい」や「楽しい」といった印象語への採点法による評価の平均評価値などを特徴量として利用できる。

5.5.2 個人属性の特徴

利用者の年齢や性別などの**個人属性情報** (demographic information) も利用できる [Pazzani 99]。これらの情報は嗜好と関連があると考えられ、マーケティングにおいてもデータベースマーケティングとして利用されてきた [Ben Schafer 01]。個人属性の特徴があれば、新規利用者に対しても推薦が可能になる利点があるが、プライバシー問題の観点から、その収集が困難である問題がある。この問題に対しては、データの利用目的を説明すること [Swearingen 01] や、14章のプライバシー協調フィルタリングの導入といった対処がある。

5.5.3 利用状況の特徴

推薦システムを利用する状況の情報である。レストランを利用する場合などでは人数や場所の情報は推薦の制約になる [山本 06, 小野 06]。また、システム側のコンテキストとして、商品の在庫や納期の情報など、推薦時には考慮すべき情報である。

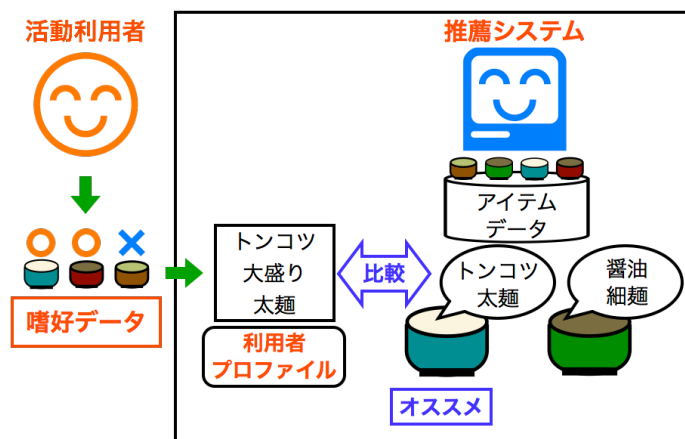
第 6 章

嗜好の予測

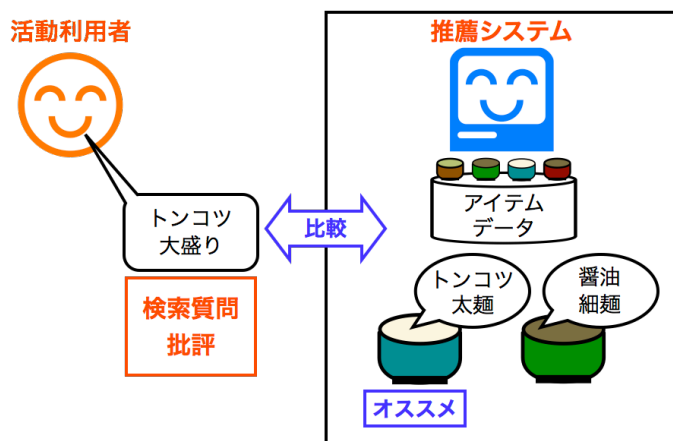
嗜好の予測とは、活動利用者の嗜好データや、アイテムの特徴を用いて、活動利用者の各アイテムへの関心や好みの度合いを予測することである。

嗜好の予測段階の実現方法は大きく二つに分類される。レンタルビデオ店で、顧客が見たい映画を推薦する場合を考えてみよう。一つは、ファンである監督、好みのジャンルを利用者に尋ねてその条件に合ったものを選ぶ方法である。これを、検索対象の内容を考慮して推薦をするので**内容ベースフィルタリング (content-based filtering)**と呼ぶ。もう一つは、映画の趣味が似ている知り合いに、面白かった映画を教えてもらう「口コミ」の過程を自動化する方法である。他の人との協調的な作業によって推薦対象を決めるため、この推薦手法は**協調フィルタリング (collaborative filtering)** や社会的フィルタリング (**social filtering**) と呼ばれている。

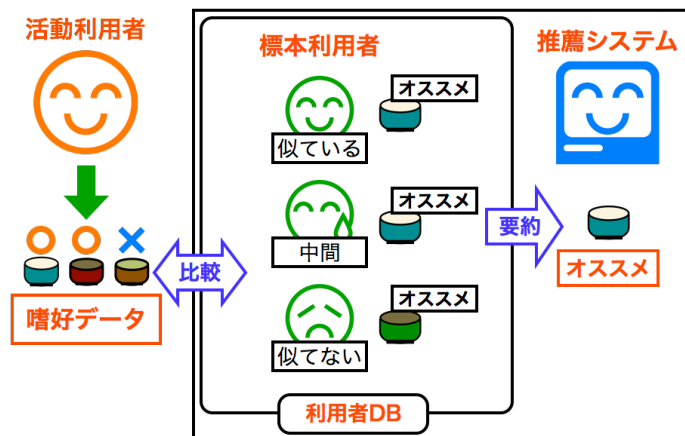
現在では、どちらの方法にもいろいろな派生型が提案されているが、純粋な形では図 6.1 のように予測する。内容ベースフィルタリング (図 6.1(a) と 6.1(b)) では、アイテムの性質と利用者の嗜好パターンを比較して、利用者が好むと判断したものを推薦する。このアイテムの性質は特徴ベクトルによって記述される。特徴ベクトルとは、アイテムのいろいろな側面の性質を表す特徴を集めて、ベクトルの形にしたものである。各特徴は、事前に定めた定義域中の値をとることで、そのアイテムの性質を表現する。ラーメンの例を示そう。このとき、スープの種類、麺の太さ、価格といった特徴から



(a) 内容ベースフィルタリング (間接指定型)



(b) 内容ベースフィルタリング (直接指定型)



(c) 協調フィルタリング

図 6.1: 内容ベースフィルタリングと協調フィルタリング

なるベクトルでラーメンを表現する。スープの種類という特徴は、トンコツ、醤油、塩のような定義域の値の一つをとり、価格という特徴では自然数とその定義域となる。そして、ある特定のラーメン『トンちゃん』があるとする

(スープの種類=トンコツ, 麺の太さ=細麺, ..., 価格= 650 円)

といったベクトルで表現される。こうした情報をいろいろなアイテムについて収集したものをアイテムデータと呼んでおく。一方、利用者の嗜好パターンは**利用者プロフィール (user profile)** によって表す。この利用者プロフィールを間接指定するシステムと、直接指定するものがある。間接指定型 (図 6.1(a)) では、利用者のいろいろなアイテムに対する嗜好データ、すなわち、好き嫌いの度合いを定量化したデータを集める。この嗜好データと、アイテムデータに基づいて、その利用者が好むアイテムの特徴のパターンを機械学習の手法でモデル化し、利用者プロフィールとする。直接指定型 (図 6.1(b)) では、利用者が明示的に、自身が好むアイテムの特徴を表した検索質問 (query) (批評 (critique) ともいう) を入力する。一般的な検索質問は「スープの種類は醤油で、価格は 500 円以下」といった、特徴に対する制約の形式だが、自然言語文などを扱えるものもある。この検索質問はそのまま利用者プロフィールとして用いられる。内容ベースフィルタリングでは、アイテムデータ中のアイテムの特徴ベクトルと、利用者プロフィールとを比較し、プロフィールに最も近い特徴ベクトルをもつアイテムを利用者が好むものと判断して推薦をする。

もう一つの協調フィルタリング (図 6.1(c)) では、アイテムの性質は全く考慮しない。その代わりに、システムが利用される前に、多くの利用者の、いろいろなアイテムに対する嗜好データをデータベースに集積している。このデータベースを**利用者データベース (user database)** (「利用者 DB」と略す) と呼び、この利用者 DB に嗜好データを登録している利用者を**標本利用者 (sample user)** と呼ぶ。さらに、今までの嗜好パターン、すなわち、どのアイテムを好み、どのアイテムを嫌うのかという傾向が類似している利用者は、これからも同じアイテムを好み、同じアイテムを嫌うであろうという仮定を導入する。この仮定の下、活動利用者と、嗜好パターンが類似している

表 6.1: 協調フィルタリングと内容ベースフィルタリングの比較

	協調	内容ベース
多様性	○	×
ドメイン知識	○	×
スタートアップ問題	×	△
利用者数	×	○
被覆率	×	○
類似アイテム	×	○
少数派の利用者	×	○

標本利用者を見つけ、これらの標本利用者が好むものを活動利用者に推薦する。

なお、内容ベースフィルタリングは、いろいろな拡張が行われているので、厳密に定義するのは難しい。文献 [Burke 02] では、個人属性の特徴 (5.5 節) などを用いず、アイテムの特徴のみを用いた、間接指定型の手法のみを内容ベースフィルタリングと定義している。そして、上記の直接指定型にあたる知識ベース型や、個人属性の特徴を使うものも、別の種類として細かく分類している。しかし、細分化しても多種多様な手法を厳密に分類するのは実際には難しいので、本稿では、広義にとらえて、アイテムや個人属性の特徴を利用するような方法は全て内容ベースフィルタリングとして扱う。一方、これらの特徴を用いず、標本利用者の嗜好データのみを用いる方法を協調フィルタリングとしておく。

第 III 部では、協調フィルタリングと内容ベースフィルタリングの各種手法を順に紹介する。さらにその後、これら二つの手法を組み合わせるハイブリッド法について述べたのち、アルゴリズムの選択の指針について述べる。これらの話題に移る前に、協調フィルタリングと内容ベースフィルタリングの長所と短所をまとめておく。

6.1 内容ベースと協調フィルタリングの比較

協調フィルタリングと内容ベースフィルタリングの長所と短所を [Balabanović 97, Burke 02] などに基づき表 6.1 にまとめた。以下、表中の各項目について詳しく述べる。

6.1.1 多様性

3.1 節で述べた多様性・セレンディピティについては、協調フィルタリングが有利と言われている。内容ベースフィルタリングでは、利用者自身が知っているアイテムの特徴に、推薦対象が制限されてしまうことが多い。例えば、映画の場合であれば、過去に見たのと同じジャンルや監督の作品が推薦される。さらに、類似した内容のアイテムを推薦し、それを利用者が受け入れることで、利用者プロファイルの偏りが一層強化される現象も生じる。それに対して、協調フィルタリングでは、自身が知らないジャンルや監督でも、他の標本利用者の知識を通じて知ることができる場合がある。そうしたときには意外性のある、すなわち、セレンディピティがある推薦ができることされている。

6.1.2 ドメイン知識

協調フィルタリングの最も重要な長所は、アイテムの特徴、すなわち、アイテムの**ドメイン知識** (domain knowledge) を全く必要としないことである。アイテムの特徴ベクトルの設計に伴う困難には次のようなことがある。

- アイテムの特徴についてのデータを集める手間やコストが必要になる。仕様についてのデータベースが整備されている商品分野は書籍、CD などに限られている。たとえ整備されていても、食品の成分表などのように推薦という目的にはあまり役立たないものもある。それ以外の分野ではデータベースの構築・更

新コストが必要になる。

- アイテムのある性質を表す特徴がないために、適切な推薦ができないことがある。例えば、映画の推薦で、自分がファンであるカメラマンの撮る映画を見る利用者がいたとしても、映画の特徴ベクトルに『カメラマン』の特徴がなければ、内容ベースでは適切な推薦ができない。
- どの特徴を採用するかということが、利用者の判断に影響を与える。推薦アイテムの決定に利用された特徴は、利用者の意志決定でより重要視され、そうでない特徴は無視されるようになる副作用を生じることがある [Häubl 01]。
- 違う分野のアイテムを推薦することが困難である。例えば、ある映画が好きでも、内容ベース法では、そのサントラの CD を推薦することは難しい。なぜなら、映画と CD は、異なる特徴ベクトルで表現されているためである。

6.1.3 スタートアップ問題

スタートアップ問題 (start-up problem) (コールドスタート問題やランプアップ問題とも呼ぶ) は、2種類に分けられる。一つは、新たにシステムを利用し始めた利用者 (first rater や early rater ともいう) に対して適切な推薦をする難しさであり、もう一つは、推薦対象として新たにシステムに導入されたアイテムを推薦する難しさである。前者については、協調フィルタリングでは、他の標本利用者との類似性を判定できないため良い推薦をするのは難しい。間接指定型の内容ベースフィルタリングでも、利用者プロフィールが不完全になるためやはり難しい。加えて、利用者本人の評価値しか利用できないので、問題がさらに深刻になることがある。だが、直接指定型の内容ベース法では、自身の利用者プロフィールを直接記述するためこうした問題は生じない。よって、観光地の案内端末での推薦など、同一利用者の継続的な利用があまりない状況では、直接指定型の内容ベース法を採用すべきである。一方、継続的な利用がなされるならば、嗜好データの蓄積に伴って予測精度が向上する協調フィルタリングや間接指定型の内容ベースフィルタリングがよいであろう。

一方、後者の新規アイテムに対する問題については、内容ベースフィルタリングが優れている。内容ベースフィルタリングでは、利用者プロフィールがあれば、新規のアイテムでも、その特徴ベクトルを手がかりに、全く問題なく推薦ができる。しかし、協調フィルタリングでは、標本利用者の評価を利用するので、まだ誰も評価していない新規のアイテムは、推薦を予測する手がかりがなく、そのアイテムに対する活動利用者の好みを判断できない。以上のことから、商品が頻繁に入れ替わるような場合は、内容ベースフィルタリングが有利である。

6.1.4 利用者数

内容ベース法の場合は、たとえシステムの利用者が一人であっても推薦は可能である。一方、協調フィルタリングは、他の利用者の意見を参照するので、利用者数がある程度なければ実行できない。また、利用者が他にいない状況では、適切な推薦もできないため、新たな利用者の増加も難しい。そのため、電子商取引サイトでのポイントの配布といったインセンティブや、暗黙的な方法で自動収集するといった手段で、十分な規模の嗜好データ量を維持する必要がある。

6.1.5 被覆率

3.1節で述べた被覆率は、適合アイテム列挙タスク（2.3節）では特に重視すべき評価規準である。協調フィルタリングでは、まだ誰も評価していないアイテムは推薦の対象にできないため、被覆率を100%にはできない。一方、内容ベースでは、アイテムの特徴を手がかりに、全てのアイテムについて推薦すべきかどうかの判定をすることができる。よって、特徴に欠損値がなければ、被覆率は100%となる。

6.1.6 類似アイテム

ドメイン知識の項では、アイテムの特徴を使うことの短所を述べたが、アイテムの特徴を使うことの長所もある。アイテムの性質を無視する協調フィルタリングでは、

同じ商品のサイズや色の違うもの、また、同じ目的の競合商品などは全く異なるものとして扱われる。例えば、利用者 A と B は共に、同じサッカーチームのファンであったとしよう。だが、A はこのチームのロゴのはいったマグカップを、B はロゴ入りのタオルを買ったとする。すると、協調フィルタリングでは、チームロゴという特性を明示的に考慮できないので、そのことを推薦に反映できない。一方、内容ベースフィルタリングでは、利用者 A が過去にマグカップとタオルを購入していれば、アイテムの特徴から、そのサッカーチームのロゴ入りの他のアイテムを推薦することも可能である。

また、推薦されたアイテムを利用者が拒否した場合、そのサイズや色が違うだけの類似アイテムを推薦されてしまう場合も、協調フィルタリングでは生じる。例えば、ある商品を却下したすぐ後で、その商品の色違いを推薦される場合はよく生じうる。これを回避するには、どのアイテムとどのアイテムを同じとみなすかが重要になる。すなわち、JAN コードなどの商品 ID が異なっても、協調フィルタリングで扱う場合には、色違いなどの類似アイテムは同じアイテムとして扱うようにする必要がある。しかし、どのアイテムを同じとみなすかは、アイテムのドメイン依存した難しい問題である。例えば、服飾などでは色の違いは重視されるだろうが、ティッシュペーパーなどは色違いでも同じアイテムとみなして良いだろう。

6.1.7 少数派の利用者

協調フィルタリングでは、非常に少数派の嗜好パターンをもつ利用者は、類似した嗜好パターンをもつ利用者を全く、または、ごく少数しか見つけられないので、適切な推薦を受けられない場合がある。例えば、ほとんど無名なタレントだが、利用者と同じ出身地であるのでファンである人がいたとしよう。こうした嗜好を持つ人は非常に希であろう。さらに、そうした人が同じシステムを利用していることはさらに希である。すると、協調フィルタリングでは、類似した嗜好の人がいなければ嗜好を予測できないので、こうした観点からの推薦は難しい。一方、内容ベースフィルタリン

グでは、タレントの出身地情報を用いて適切な推薦をすることも可能である。

第7章

推薦の提示

本章では、4章の推薦システムの実行過程の最後の段階である「推薦の提示」についてまとめる。この段階では、予測した評価を、活動利用者の目的に適した形式で提示する。なお、推薦システムの利用者へのアンケート調査結果に基づいて、結果の出力に対する設計指針を、定性的な面から論じた文献に [Swearingen 01] がある。推薦したアイテムに関する情報も提示すべきことや、推薦リストのレイアウトや、これを閲覧するインターフェースの重要性などを指摘している。

7.1 推薦の配送

利用者に推薦結果を届ける手段は、次のように分類できる。

push 型 利用者が、システムを直接には使っていない場合に、利用者に推薦を届ける。

個人化していないバナー広告や、個人化したメールマガジンなどが該当する。

pull 型 システムを使用中の利用者が要求したときに、それに応じて推薦結果を届ける。検索質問を使う直接指定型の内容ベースフィルタリングの推薦は、この方法を用いる。

passive 型 利用者がシステムを使用中に、推薦結果を添付しておく。例えば、商品と共に推薦の度合いを★印で示したり、関連商品を同時に示したりする。利用者

に主導権があり、システム側は積極的な推薦はしない。

2.2 節の 5 種類の運用側の目的のうち「通知サービス」では push 型を、他の目的では pull 型か passive 型を利用することになる。

その他、配送するかどうかの決定についての工夫もある。マイクで獲得した発話内容から、視聴中の TV 番組への関心の度合いを推定し、関心が低いと推定されたときにのみ積極的に推薦をする方法 [高間 07] などが提案されている。

7.2 推薦アイテムの選別

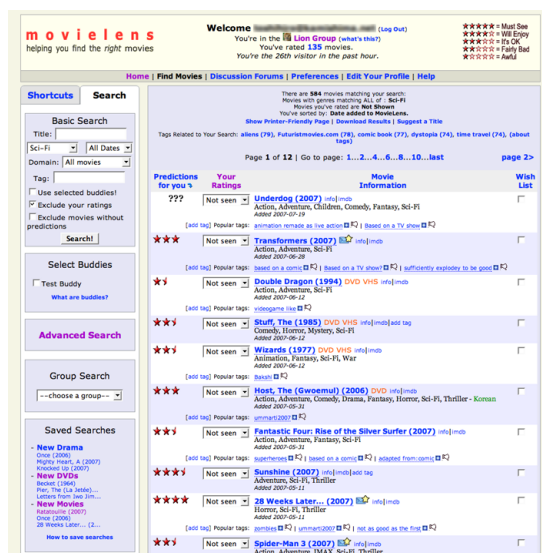
予測評価の高いアイテムでも、推薦すべきでないアイテムがある。そういったものを必ず選別し除外しておく必要がある。

利用者が既知であると分かっているアイテムを推薦してもほとんど意味がないので、これらのアイテムを除外する。例えば、同じセッション中ですでに提示したアイテムや、購買履歴から過去に購入したことがあるものなどを除外する。また、非個人化推薦である売上げランキングなどを同時に表示する場合は、重複したアイテムを除外しておく方がよいだろう。文献 [清水 07] では、利用者から、アイテムへの好き嫌い評価だけでなく、既知かどうかの情報も得る。その情報に基づいて、利用者がアイテムを既知かどうかを予測し、未知と予測されるアイテムをより積極的に提示する。

その他、アイテム、個人属性、およびコンテキストの情報に基づいた除外も必要になる。例えば、アイテムに依存した条件としては、在庫がないとか、利用者が海外在住で発送できない理由で提供できない商品がある。当然ながら、違法なアイテムも除外すべきである。また、色違いなど差異がわずかなものは、代表的なものをひとつだけ残して、残りは提示しないといったことを行う。個人属性の特徴は、女性専用の旅行プランは男性には推薦しないといったことに利用できる。また、コンテキストの情報に関しては、利用者の現在位置から遠いレストランは推薦すべきではないし、夏に冬物衣料をすべきではないといった季節商品の問題もある。



(a) 適合アイテム発見 (Amazon.co.jp)



(b) 評価値予測 (MovieLens [MovieLens])

図 7.1: 推薦結果の表示例

スクリーンショットは 2007/07/26 に取得した。

7.3 推薦の表示形式

推薦すべきアイテム群を表示する形式は、2.3 節のタスクの種類に応じて以下のよう
に適切なものを選択すべきである。

適合するものを一つ見つけ出す適合アイテム発見を目的とする場合には、予測され
る評価の高いものから順に整列したリストを利用者に提示するのが一般的である (図
7.1(a))。利用者はこのリストを上位から閲覧することで、自分の嗜好に適合したアイ
テムを素早く見つけることができる。

評価値予測で閲覧中にアイテムの評価を得ることを目的とする利用者は、積極的な
決定をする意図をもっていない。そこで、閲覧中のアイテムに、活動利用者の予測評価
値を付随的に提示する。これは、★の数や、アイコン、グラフなどで表す (図 7.1(b))。
このような情報を参照することで、多数の候補の中から、利用者にとって関心のある
ものを中心に閲覧できるようになる。

7.4 多様性の向上

評価値予測 (2.3 節) を目的とする場合は、利用者の関心の幅を広げるため、多様性 (3.1 節) の高い推薦が望まれる。そこで、文献 [Ziegler 05] では多様性を向上させる目的で、予測精度を犠牲にしても、より広範囲の分野のアイテムを推薦する **話題多様化** (topic diversification) を提案している。この多様化の有効性を、同種のアイテムに対しては、1 度目より 2 度目、2 度目より 3 度目に支払いたいと思う対価が減少する経済学における収穫逓減の法則 (law of diminishing marginal returns) との関連付けて論じている。具体的には、アイテムの階層的な分類を導入し、分類階層の近さによってアイテム間の類似度を測る。純粋に予測精度を重視したリストの上位から順に、最終推薦リストにアイテムを一つずつ追加するが、このとき、すでに推薦したアイテムと類似しているアイテムは推薦されにくいようにする。この方法により、予測精度を犠牲にして、アイテムの多様性をより重視するような推薦をする実験を行った。予測精度を単調に減少させ、徐々に多様性を高めると、最初は利用者は推薦の多様性の高まりを認知できたが、ある程度以上になると認知できなくなった。よって、推薦リストの多様性を利用者が認知できる程度にとどめれば、予測精度もそれほど低下せず、利用者の満足は高まると報告している。その他、推薦リストに、新製品やあまり知られていないアイテムを必ず混ぜるといった手法も考えられる [Swearingen 01]。

7.5 推薦理由

利用者は、不必要に高価なものを薦められていると疑ったりするため、推薦したアイテムを必ずしも採用するわけではない。採用されない推薦は無意味なので、推薦ができるだけ採用されるような工夫が必要である。そうした工夫として、アイテムの **推薦理由** (explanation of recommendations) も示すことが有効とされている。

文献 [Sinha 02] では、推薦の **透明性** (transparency) と利用者の推薦結果に与える印象との関係を調査している。ここでいう透明性とは、利用者の入力した評価やその他

の情報と、出力された推薦との間の因果関係が明確に説明されていることである。この調査では、12人の被験者に5種類の商用音楽推薦システムを利用させた。そして、推薦されたアイテムを好むか、推薦は信頼できるか、そして推薦に透明性があったかの質問をした。調査の結果は、透明性があると考えた場合の方が、そうでない場合に比べて、有意に推薦された結果を好み、また、その結果を信頼できると答えた。さらに、推薦されたアイテムを、(a)知らない場合、(b)知っていてかつそれを好きな場合、および(c)知っていてかつそれを嫌いな場合に分けた。推薦されたアイテムを好むかどうかの質問については、(b) (a) (c)の順に好んだ。さらに、(a)と(b)の推薦については、推薦に透明性があると、ないときよりも、有意に推薦されたアイテムを好んだ。しかし、(c)の推薦については、有意な差はなかった。これらのことから、利用者が推薦に透明性があると考えたときには、推薦されたものをより好み、その推薦を信頼する。また、知っている好きなアイテムを推薦されると、推薦に透明性があると利用者は考えるといえる。なお、実際がどうであるかにかかわらず、透明性があると考えたかを利用者に質問した結果であること、また、システムがデータをねつ造していない点については、利用者はシステムを信頼していることには注意されたい。利用者もおそらく知っているであろう、一般に知られたアイテムを推薦して、利用者の推薦への信頼を向上させるといったこともできる [Swearingen 01].

推薦の透明性を高める、すなわち、利用者が入力した嗜好データから、推薦を導いた根拠を明示することで、推薦への信頼を高める試みがある。文献 [Herlocker 00] では、こうした根拠を提示する手法を比較している。その中で最も有効な方法とされたのが図 7.2(a)の方法である。これは、活動利用者と類似した嗜好をもつ他の利用者の、推薦した映画についての評価値を棒グラフで示したものである。この例では、推薦した映画について、嗜好が似ている人たちのほとんどが、最高か、それに次ぐ評価をしていることが、一目で分かる。このように、推薦の手順の内部を示すアプローチを**ホワイトボックス (whitebox)** アプローチと呼ぶ。その次に有効であったのは、そうした手順を示さない**ブラックボックス (blackbox)** アプローチの、推薦の確信度を示すことであった。ここで、推薦の**強度 (strength)** と**確信度 (confidence)** について述べておこ



(a) 類似した嗜好の利用者の評価の提示 [Herlocker 00] (b) 推薦の根拠となった嗜好データの提示 (Amazon.co.jp)

図 7.2: 推薦理由の提示例

スクリーンショットは 2007/07/26 に取得した。

う。推薦の強度とは、どれくらいアイテムを好むと予測しているかということである。利用者の嗜好を 5 段階で予測するなら、5 と予測したときに強い推薦といえる。一方の推薦の確信度とは、この推薦をどれくらい確かだと考えているかということである。10.3 節の確率モデルで、適合/不適合の 2 段階で活動利用者の嗜好を予測したとしよう。このとき、適合すると予測した確率が 0.6 でも 0.9 でも、予測は「適合」だが、後者のときにより確かな推薦といえる。さらに、他の方法でも実験したが、利用者の満足度を向上させた方法は無かったと報告している。

その他、9.2 節のようにアイテム間の類似度に基づく推薦では、推薦に関連したアイテムを提示するというホワイトボックスアプローチもある。図 7.2(b) は、活動利用者自身の、推薦の根拠となった他のアイテムへの評価を提示したものである。また、暗黙的に嗜好データを獲得した場合、推薦の根拠を利用者が認知できなくなるので、推薦と共に根拠となった嗜好データを示す方法もある。ただし、予測評価値などを提示して、利用者が嫌いなアイテムが高評価されていると、システムに対する信頼を失う危険性も指摘されている [Swearingen 01].

第 III 部

推薦システムのアルゴリズム

第 8 章

協調フィルタリング

ここでは、協調フィルタリングによる嗜好の予測について述べる。ここでいう予測とは、活動利用者はまだ知らないが、他の標本利用者は知っているアイテムについて、活動利用者の関心の有無や、評価値を推定することである。このように、与えられたデータの中から、規則性を見つけ出し、その規則性に基づいて予測する問題は、機械学習や統計的予測の枠組みによって解く。だが、万能な予測手法は理論的にありえないことは、ノーフリーランチ定理として知られている。よって、3章で述べた、利用者数やアイテム数などのデータの特性、望ましい推薦が備えるべき性質を考慮してモデルやアルゴリズムを選択しなくてはならない。もしこれらが適切でなければ、データがいくらあっても予測精度が向上することはないので、この選択は重要である。これらの選択に関連した、モデルやアルゴリズムの特徴に留意して述べるので、参考にされたい。

8.1 メモリベース法とモデルベース法

個々のアルゴリズムについて述べる前に、協調フィルタリング手法の分類と、それぞれのグループの長所と短所を述べる。推薦候補の予測手法は**メモリベース法** (memory-based method) (事例ベース法 (instance-based method) ともいう) と**モ**

表 8.1: メモリベース法とモデルベース法の比較

	メモリベース法	モデルベース法
推薦時間	×:遅い	○:速い
適応性	○:あり	×:なし

モデルベース法 (model-based method) に分けられる [Breese 98]. メモリベース法では、推薦システムが利用される以前には何もせず、ただ利用者 DB を保持している。そして、推薦をするときに、利用者 DB 中の嗜好データそのものと、活動利用者の嗜好データを併せて予測をする。もう一方のモデルベース法では、推薦システムが利用される以前に、あらかじめモデルを構築する。このモデルは、「佐藤さんが好むものは、鈴木さんも好むことが多い」といった、利用者とアイテムの嗜好についての規則性を表す。推薦をするときに、利用者 DB は用いずに、このモデルと活動利用者の嗜好データとに基づいて予測する。すなわち、事前にモデルを構築するかどうかという違いが重要である。

これら二つの手法の大まかな長所と短所を表 8.1 にまとめておく。推薦時間についてだが、メモリベース法は一般に遅い。これは、利用者 DB には多数の標本利用者や推薦対象が登録されており、これら多数の項目を推薦の度に調べ直すのに時間がかかるためである。一方のモデルベース法では、夜間などシステムが利用されない間や、毎月や毎週など定期的にモデルをあらかじめ構築・更新しておく。すると、この時間は推薦の時間からは除外できる。また、モデルの規模は、利用者 DB のそれと比べて小さいので、活動利用者に速く推薦することができる。

表 8.1 の適応性とは、標本利用者や推薦対象が、削除されたり追加されたりしても、適切な推薦ができるかということである。このような削除や追加をすると、モデルベース法では、利用者や推薦対象間の規則性に変化が生じるため、モデルを再構築する必要が生じる。だが、モデルの構築には時間がかかるので、頻繁に行うことは難しい。そのため、適応性に関してはモデルベース法は不利になる。一方、メモリベース

法では、モデルの構築は行わないためこのような問題は生じない。

第 9 章

メモリベース型協調フィルタリング

メモリベース法 (memory-based method) とは、利用者 DB を直接利用して、活動利用者の嗜好を推定する方法である。機械学習の観点からは k 近傍法 (k -nearest neighbor method) とみなせる。この方法は、大きく**利用者間型メモリベース法** (user-user memory-based method) と**アイテム間型メモリベース法** (item-item memory-based method) とに分類できる。どちらも、類似性を計算する点では共通しているが、利用者間型は、活動利用者と嗜好パターンが似ている利用者をまず見つけ、彼らが好むものを推薦する。一方、アイテム間型では、活動利用者が好むアイテムと類似したアイテムを推薦する。これらの手法を順に紹介する。

9.1 利用者間型メモリベース法

利用者間型メモリベース法の代表的な手法である GroupLens の方法 [Resnick 94] について述べる。これは、協調フィルタリングの手順を自動化した先駆的研究である。簡潔な手法だが、その予測精度は高く、多くの改良研究もなされている。GroupLens は、当初は NetNews の記事を推薦するシステムであったが、現在は映画の推薦シス

テム MovieLens となっている。この実験システムは、プロジェクトのホームページ [MovieLens] で公開されているので、推薦システムを体感するため利用してみるとよいだろう。

レストランを探す場合のことを考えてみよう。このとき、自分と食べ物の嗜好が似ている何人かの人に尋ねてみて、彼らの意見をもとにどの店で食事をするか決めたりすることがあるだろう。GroupLens の方法は、こうした人のコミュニティでの「口コミ」による推薦の過程を、次の二段階で実現する。

類似度の計算 利用者 DB 中の各標本利用者と活動利用者の嗜好の類似度を求める。類

似度とは、嗜好パターンがどれくらい似ているかを定量化したものである。

嗜好の予測 活動利用者は知らないアイテムについて、それらのアイテムへの標本利用

者の好みと、その標本利用者との間の類似度に基づいて、活動利用者がどれくらいそのアイテムを好むかを予測する。

これらの段階について詳細に説明する前に、記号を定義する。 n 人の全利用者の集合を $\mathcal{X} = \{1, \dots, n\}$ 、 m 種類の全アイテムの集合を $\mathcal{Y} = \{1, \dots, m\}$ とする。評価値行列 \mathbf{R} は利用者 $x \in \mathcal{X}$ の、アイテム $y \in \mathcal{Y}$ への評価値 r_{xy} を要素とする行列である。 r_{xy} は、評価済みなら評価値の定義域 \mathcal{R} のいずれかの値をとり、未評価なら欠損値「 \perp 」をとる。例えば、5 段階のスコアを用いた採点法 (5.2 節を参照) で獲得した嗜好データであれば、評価値の定義域は $\mathcal{R} = \{1, \dots, 5\}$ となる。活動利用者を添え字 a で表す、すなわち、 r_{ay} は活動利用者のアイテム y への評価値である。また、利用者 x が評価済みのアイテムの集合を $\mathcal{Y}_x = \{y \mid y \in \mathcal{Y}, r_{xy} \neq \perp\}$ で表す。活動利用者と標本利用者 x の類似度は、共通に評価しているアイテムについての Pearson 相関で測る。

$$\rho_{ax} = \frac{\sum_{y \in \mathcal{Y}_{ax}} (r_{ay} - \bar{r}'_a)(r_{xy} - \bar{r}'_x)}{\sqrt{\sum_{y \in \mathcal{Y}_{ax}} (r_{ay} - \bar{r}'_a)^2} \sqrt{\sum_{y \in \mathcal{Y}_{ax}} (r_{xy} - \bar{r}'_x)^2}} \quad (9.1)$$

ただし、 \mathcal{Y}_{ax} は利用者 a と x が共通に評価したアイテムの集合、すなわち、 $\mathcal{Y}_{ax} = \mathcal{Y}_a \cap \mathcal{Y}_x$ で、 $\bar{r}'_x = \sum_{y \in \mathcal{Y}_{ax}} r_{xy} / |\mathcal{Y}_{ax}|$ である。なお、 $|\mathcal{Y}_{ax}| \leq 1$ 、すなわち、活動利用者 a と標本

表 9.1: 評価値行列 \mathbf{R} の例

	1: 親子丼	2: 牛丼	3: 海鮮丼	4: カツ丼
1: 山田	1	3	⊥	3
2: 田中	⊥	1	3	⊥
3: 佐藤	2	1	3	1
4: 鈴木	1	3	2	⊥

利用者 x が共通に評価したアイテムが一つ以下ならば、Pearson 相関は計算できないので $\rho_{ax} = 0$ とする。アイテム $y \notin \mathcal{Y}_a$ の評価値は、式 (9.1) の類似度で重み付けした、各標本利用者のアイテム y への評価値の加重平均で予測する。

$$\hat{r}_{ay} = \bar{r}_a + \frac{\sum_{x \in \mathcal{X}_y} \rho_{ax} (r_{xy} - \bar{r}_x)}{\sum_{x \in \mathcal{X}_y} |\rho_{ax}|} \quad (9.2)$$

ただし、 \mathcal{X}_y はアイテム y を評価済みの利用者の集合で、 \bar{r}_x は利用者 x の全評価アイテムに対する平均評価値 $\bar{r}_x = \sum_{y \in \mathcal{Y}_x} r_{xy} / |\mathcal{Y}_x|$ である。なお、活動利用者がアイテム y を評価済みである $a \in \mathcal{X}_y$ である状況では、そもそも \hat{r}_{ay} の推定が不要になるので想定しなくてよい。この式の第1項は、第2項が中間的な評価で0をとるので、それを補正するバイアス項である。また、第2項の分子は上記の加重平均であり、分母は評価している利用者が多い、すなわち $|\mathcal{X}_y|$ が大きいと加重平均が大きくなりやすい問題に対する正規化項である。また、文献 [Resnick 94] では、式 (9.2) の第2項のように、活動利用者と標本利用者が共に評価したアイテム \mathcal{Y}_{ax} 上での評価値の平均 \bar{r}'_x を用いている。だが、現在の GroupLens プロジェクトでは、単純に評価済みアイテムの平均値 \bar{r}_x を用いているとのことであり^{*1}、また、筆者の実験でも、どちらを用いても有意な差はみられなかった。加えて、 \bar{r}_x の方が、活動利用者が決まる前に事前に計算できる利点があるので、実際に利用するときには \bar{r}_x を用いる方が良いだろう。

ここで簡単な例を挙げよう。表 9.1 はどんぶり専門店『丼兵衛』の評価値行列 \mathbf{R} の

^{*1} J. Riedl のメールより

例である。4人の利用者は各行に対応し、4種類のアイテム(=どんぶり)は列に対応する。評価値は3段階の採点法、すなわち、 $\mathcal{R} = \{1, 2, 3\}$ である。この表は、利用者3の佐藤のアイテム2の牛丼の評価値は1で、嫌いであることを示している。具体的に、活動利用者を田中、すなわち $a = 2$ とし、2:田中の1:親子丼への推定評価値 $\hat{r}_{2,1}$ を求めてみよう。

まず、式(9.1)の相関係数を求める。親子丼を評価済みの利用者、すなわち \mathcal{X}_1 に含まれる各利用者と活動利用者との間の相関係数を求める必要がある。1:山田、3:佐藤、4:鈴木の3人とも親子丼を評価済みなので $\mathcal{X}_1 = \{1, 3, 4\}$ の各利用者との相関係数を求める。2:田中と1:山田の相関係数 $\rho_{2,1}$ は、共通に評価しているアイテムが2:牛丼だけで、1個以下なので $\rho_{2,1} = 0$ となる。次に、2:田中と3:佐藤の間の相関係数を計算する。この二人が共通に評価しているのアイテムは2:牛丼と3:海鮮丼なので $\mathcal{Y}_{2,3} = \{2, 3\}$ となる。すると、これらのアイテムについての $\mathcal{Y}_{2,3}$ 上の平均評価値はそれぞれ

$$\begin{aligned}\bar{r}'_2 &= \left(\sum_{y=2,3} r_{2,y} \right) / 2 = (1 + 3) / 2 = 2 \\ \bar{r}'_3 &= \left(\sum_{y=2,3} r_{3,y} \right) / 2 = (1 + 3) / 2 = 2\end{aligned}$$

となり、相関係数は次式になる。

$$\begin{aligned}\rho_{2,3} &= \frac{\sum_{y=2,3} (r_{2,y} - \bar{r}'_2)(r_{3,y} - \bar{r}'_3)}{\sqrt{\sum_{y=2,3} (r_{2,y} - \bar{r}'_2)^2} \sqrt{\sum_{y=2,3} (r_{3,y} - \bar{r}'_3)^2}} \\ &= \frac{(1-2)(1-2) + (3-2)(3-2)}{\sqrt{(1-2)^2 + (3-2)^2} \sqrt{(1-2)^2 + (3-2)^2}} \\ &= 1\end{aligned}$$

同様に計算すると2:田中と4:鈴木の相関は $\rho_{2,4} = -1$ となる。

次に推定評価値を計算する。まず、2:田中の全評価済みアイテム上の平均評価値を求める。

$$\bar{r}_2 = \left(\sum_{y=2,3} r_{2,y} \right) / 2 = (1 + 3) / 2 = 2$$

最後に、これまで計算した値を式 (9.2) に代入すると

$$\begin{aligned}\hat{r}_{2,1} &= \bar{r}_2 + \frac{\sum_{x=1,3,4} \rho_{2,x} (r_{x,1} - \bar{r}'_x)}{\sum_{x=1,3,4} |\rho_{2,x}|} \\ &= 2 + \frac{0(1-3) + 1(2-2) + (-1)(1-5/2)}{|0| + |1| + |-1|} \\ &= 2.75\end{aligned}$$

によって、2:田中の1:親子丼への推定評価値は2.75と計算できる。この値は、最大値3にかなり近く、2:田中は1:親子丼が好きであると予測される。

9.1.1 利用者間型メモリベース法の改良

GroupLensの方法にはいろいろな改良が試みられている。文献 [Herlocker 99] では、いろいろな改良を実験的に検証している。全ての計算の前に、評価値 r_{xy} から利用者 x の平均評価値 \bar{r}_x を引いて正規化しておくことで予測精度は向上する。これは、5.2節で述べたように、計測された評価値揺らぎや偏りがある。肯定的でも否定的でもない評価値を0に正規化することで、こうした不整合が緩和されるためであろうと著者は考える。利用者の近傍を使う改良もある。ここでの近傍とは、式 (9.1) の相関が大きな、すなわち活動利用者と類似した嗜好をもつ利用者の集合のことである。式 (9.2) の推定評価値は、アイテム y を評価済みの全ての利用者の評価に基づいているが、これを事前に計算した活動利用者の近傍のみに基づいて計算する。実験によれば、近傍利用者数がある程度以上になると、それ以降は増やしても予測精度は向上しない。よって、近傍利用者だけに計算を限定することで計算量を減らし、データベースの参照も抑制できるので、効率よく計算できるようになる。ただし、モデルベース法のモデルほど頻繁にする必要はないが、近傍は定期的に更新しなくてはならないので、純粋なメモリベース法の利点は部分的には失われる。また、新規の参加者については近傍を新たに計算する必要が生じ、脱退者が他の利用者の近傍利用者であれば予測精度の低下を招く。以上の、二つの改良は、ほとんどのデータに対して有効なので、適用しておく方がよいだろう。その他、この文献 [Herlocker 99] では、データが疎である問題

に対処するため、活動利用者と標本利用者が共通に評価しているアイテムの数に応じて重み付けしたり、式 (9.1) の Pearson 相関の代わりに順位相関を使ったりすることは予測精度の向上に役立ったと報告している。

文献 [Breese 98] にも、いくつかの改良が示されている。その一つに、評価値の数が少ない場合に有効な**デフォルト投票** (default voting) がある。利用者間の類似度は、共通に評価しているアイテム \mathcal{Y}_{ax} への評価に基づいて計算する。しかし、そうしたアイテムが少ないか、全くない場合には、適切に類似度を評価できない。例えば、利用者 1 と 2 の評価済みアイテム集合がそれぞれ $\mathcal{Y}_1 = \{1, 2, 3\}$ と $\mathcal{Y}_2 = \{1, 4, 5\}$ とする。このとき、これら二人の類似度は共通して評価しているアイテム 1 に対する評価値 $s_{1,1}$ と $s_{2,1}$ の二つの値だけに依存する。このとき類似度は、共通に評価したアイテム数 $|\mathcal{Y}_1 \cap \mathcal{Y}_2|$ は一つだけなので 0 となってしまう。そのため、これらの利用者間の類似度は全く推薦の精度向上に寄与しない。そこで、利用者 1 の $r_{1,2}$, $r_{1,3}$ と、利用者 2 の $r_{2,4}$, $r_{2,5}$ の評価値を活用する。例えば、 $r_{1,2}$ の評価値は、利用者 2 の対応する値 $r_{2,2}$ があれば相関の計算に利用できる。そこで、アイテム 2 の中立的な評価値、すなわち、全利用者のアイテム 2 への評価値の平均 $\bar{r}_y = \sum_{x \in \mathcal{X}} r_{xy} / |\mathcal{X}_y|$ をデフォルト投票値とし、 $r_{2,2}$ の値をこのデフォルト投票値で置き換える。他のアイテムについても同様の置き換えをすると、アイテム集合 $\mathcal{Y}_1 \cup \mathcal{Y}_2 = \{1, 2, 3, 4, 5\}$ 上でより適切に利用者間の類似度を評価できるようになる。筆者の実験でも、一人あたりの評価値数が少ない状況でかなりの効果が確認された。この文献 [Breese 98] では、最低や最高など両端に近い評価値をより重視する改良や、評価している利用者が少ないアイテムへの評価値を重視する改良などを提案している。

3.2 節で述べたように推薦システムが扱うデータは非常に疎である。機械学習では、このような場合には、次元削減の前処理を実行することが多い。代表的な方法に主成分分析 [Bishop 06, Bishop 08, 石井 98] がある。GroupLens の方法では、各利用者は m 個のアイテムへの評価値を要素とするベクトルで表される。これをデータの分散に関する情報をできるだけ失わないように、 $K < m$ の K 要素のベクトルで各利用者を表現するのが主成分分析である。この K 要素に情報を圧縮したベクトルで、利用者間

の類似度を求める方法もよく用いられている [Canny 02a, Zhang 06b]. ただし, 主成分分析は欠損値があると実行できないので, 上記のデフォルト投票値や EM アルゴリズムなどによって適当に評価値行列 S を補完する必要がある.

9.2 アイテム間型メモリベース法

GroupLens などの利用者間型では, 評価値行列 \mathbf{R} の行ベクトル, すなわちある利用者のいろいろなアイテムへの嗜好を表すベクトルの類似度に基づいて, 他の利用者の評価値を予測した. 一方, アイテム間型メモリベース法では, アイテムベクトル, すなわち, \mathbf{R} の列ベクトルの類似度を用いる. これは, いろいろな人に同じような評価を受けるアイテムは似ていると考え, ある利用者が関心をもっているアイテムの類似アイテムにも, その利用者は関心をもつという仮定に基づいている. 簡単な方法としては, アイテムベクトルのコサイン [Linden 03] や, 単純な共起性 [Das 07], Pearson 相関などでアイテム間の類似度を測り, 利用者が閲覧中や, 買い物かごに入れている, もしくは, 直近の利用履歴にあるアイテムと類似しているアイテムを推薦する. これら方法によって, 協調フィルタリングの枠組みで, 一時的個人化 (2.1 節) をした推薦ができる.

GroupLens の方法と同様に加重平均を使う方法としては [Sarwar 01] がある. この方法では, 活動利用者のアイテム y への推定評価値を次式で求める.

$$\hat{r}_{ay} = \frac{\sum_{j \in \mathcal{Y}_a} \rho'_{yj} r_{aj}}{\sum_{j \in \mathcal{Y}_a} |\rho'_{yj}|} \quad (9.3)$$

ただし, ρ'_{yj} はアイテム y と j の類似度である. この方法は文献 [Ali 04] でも利用され, アイテム間の類似度行列と活動利用者自身の評価値があれば利用者 DB を参照することなく, ローカルマシンだけで推薦を計算できる. よって, ある程度のプライバシー保護 (14 章) も実現できる. GroupLens の方法と同様に, \mathcal{X}_a に含まれるアイテム全てではなく, ρ'_{kj} がしきい値以上の近傍に計算を限定すると, 予測精度を下げることなく, 式 (9.3) の計算を高速化できる. だが, 近傍アイテムが重要なので, 公開中

の映画・コンサートなどのように推薦対象のアイテムが頻繁に入れ替わる **item churn** という状況では、頻繁な近傍の更新が必要になってしまう。この方法は、評価しているアイテム数が少ない利用者には推薦が早くできる。利用者間型より予測精度はよいとの報告もある。しかし、理論的根拠はないが、実験的には特定のアイテムに推薦が偏る傾向が強く [McNee 06a]、多様性についてはアイテム間型は不利であるといわれている。

9.3 メモリベース法に関するその他の研究

5.2 節でも述べたように、利用者の評価値には一貫性が低い問題がある。そこで、McLaughlin ら [McLaughlin 04] は、投票された評価値を隣接した評価値に配分する信念分配 (belief distribution) を利用した GroupLens 法の拡張を示した。利用者が指定した評価値に最も大きな重みを与えると共に、その周囲の評価値にも小さな重みの評価値を与える。例えば、評価値 4 を利用者が指定したとき、評価値 4 には 0.6 の重みで、それに隣接する評価値 3 や 5 には 0.2 の重みがあると考え。すなわち、ファジィ理論でのメンバーシップ関数のようなものである。しかし、この重みの分配の割合は予測精度に大きく影響するが、この割合は調整は試行錯誤によって行わなければならない問題がある。

5.2 節では、利用者から嗜好データを得るために、採点法や格付け法ではなく順位法を用いるなんとなく協調フィルタリング [Kamishima 03b] について述べた。ここでは、順位法で得た嗜好順序を使って推薦する方法について述べる。手法は非常に単純で、嗜好順序中のアイテムの順位をそのまま評価値とする。例えば、利用者 x がアイテム 1, 2, 3, 4 を好きなものから順に $3 > 2 > 4 > 1$ と並べた場合、アイテム 4 の順位は 2 であり、 $r_{x4} = 2$ として扱う。このとき、 r_{xy} が小さいほど好きなことを表すので、式 (9.2) の \hat{r}_{ay} が小さいものから順に推薦する。また、順位法では評価は常に相対的なもので、予測評価値も相対的な好みしか示さないことに注意しなくてはならない。利用者ごとに整列したアイテム数が異なる場合には、嗜好順序の長さが一定ではない。この

場合は、嗜好順序の長さを L とし、この嗜好順序中のアイテム y の順位を rank_y としたとき、 $r_{xy} = \text{rank}_y(m+1)/(L+1)$ とする。これは、嗜好順序に含まれる L 個の対象が、 m 個の全対象から一様にランダムに選ばれたとしたときの、アイテム y の順位の期待値である。著者の実験 [Kamishima 06] では、採点法で得た評価値を正規化するなどしても、順位法による嗜好データに基づく予測順序の精度の方が良かった。

GroupLens の方法では、利用者間の類似度は式 (9.1) の Pearson 相関で測っている。実験的にはこの類似度でかなり良い予測精度が得られているが、理論的な根拠は弱い。そこで、さらに予測精度を向上させるため、経験損失を最小にするようにこの類似度関数を学習する方法 [Bell 07b] もある。これにより予測精度は向上するが、8.1 節で述べたモデルベース法と同じ適応性の問題を生じてしまう。

純粹に理論的な観点から、Pennock ら [Pennock 00b] は、望ましい協調フィルタリングが備えるべき公理的性質について論じた。これは、社会全体での意志決定を論じる社会選択の研究で著名な Arrow の不可能性定理 [Arrow 70] のような考え方である。四つの公理的性質として、全定義域と最小機能 (universal domain and minimal functionality)、全員一致 (Pareto 性, unanimity)、無関係な候補からの独立性 (independence of irrelevant alternatives)、スケール不変性 (scale invariance) を挙げている。これらの、性質を満たす予測手法は最近傍法のみであることを示している。実際の推薦システムで、これらの公理的性質が厳密に満たされなければならないわけではないが、こうした研究は推薦システムの手法の選択指針について参考になるであろう。

第 10 章

協調フィルタリング：モデルベース法

モデルベース法では，活動利用者に推薦をする前にモデルを構築する．ノーフリーランチ定理により万能な予測モデルは存在しないので，多様なモデルが目的に合わせて提案されている．これらを順次紹介する．

10.1 クラスタモデル

嗜好パターンが類似している利用者が好むものを推薦するという手順を，直観的に実装したのがクラスタモデルである [Breese 98, Kamishima 03b]．クラスタとは対象の集合を分割した部分集合で，同じクラスタ内の対象は互いに似ているが，違うクラスタでは似ていないという条件を満たすものである．こうしたクラスタを得る手法をクラスタリングという [神嶋 03a, 齋藤 06, 宮本 99]．この手法を用いて，嗜好パターンが類似している利用者のクラスタに，標本利用者の集合を分割する．ここで，利用者間の類似度は，9 章の式 (9.1) の Pearson 相関係数のように，いろいろなアイテムへの評価値がどれくらい類似しているかによって測る．活動利用者への推薦は，活動利用者と各クラスタとの類似度を調べ，最も似ているクラスタを見つける．そして，

そのクラスタ中の標本利用者の平均評価値が高いアイテムから順に活動利用者に推薦する。この方法には、利用者集合とアイテム集合を同時に分割する共クラスタリング (co-clustering) を使う改良 [George 05] などがある。

このモデルでは、利用者 DB を何個のクラスタに分けたかによって、推薦の質が大きく変わる。クラスタ数を小さく設定すると、おおまかであまり個人化されていない推薦がなされる。そのため、cold-start 問題に対して比較的強い。しかし、推薦パターンの種類はたかだかクラスタ数に制限されるので、多数の嗜好データを集めても、ある程度以上に個人化された推薦はできない。一方、クラスタ数を多くすると、スタートアップ問題に対して弱くなるが、推薦の個人化の度合いは高まる。また、クラスタ数が多すぎると安定したクラスタを求めるのが難しくなる問題もある。よって目的に合わせてクラスタ数を調整する必要がある。

この方法には、実現が直観的なことに加え、モデルの構築も比較的高速であり、単純なので実装も容易であるといった利点がある。推薦時も、各クラスタと活動利用者の類似性を調べるだけなので、計算量はクラスタ数だけに比例し、高速である。欠点としては**灰色の羊問題** (gray sheep problem) [Burke 02] がある。例えば、映画の推薦を考えた場合、特定のジャンルのものだけを鑑賞する利用者はむしろ少なく、サスペンスとホラーなど複数のジャンルの映画を見るであろう。クラスタモデルでは、利用者を特定のクラスタに分けてしまうため、こうした白でも黒でもない灰色の羊の利用者に適切な推薦ができない。

10.2 関数モデル

利用者が好きなものほど大きな値をとる効用関数 (utility function) や、評価値そのものを予測する関数を用いるモデルを、本稿ではまとめて関数モデルと呼ぶ。そして、これらのモデルの獲得を、回帰問題、クラス分類問題、および順序回帰問題に帰着させて解く。これらの手法を順次紹介する。

10.2.1 回帰問題に帰着させる方法

最初に、回帰問題に帰着させる手法から紹介する。最も簡単な線形関数の場合を考える。9.2 節の式 (9.3) は、詳細を無視すれば次のような線形モデルとみなせる。

$$\hat{r}_{ay} = \sum_j w_{yj} r_{aj} \quad (10.1)$$

9.2 節では、パラメータ w_{yj} はアイテム y と j の嗜好パターンの類似性を、相関係数など経験的に選んだ類似度で決めた。しかし、与えられた評価値の集合を訓練事例として、機械学習の手法を適用すれば、もっと予測精度の高い関数を獲得できるのではないか？ この考えに従い、これらのパラメータを決める問題を**回帰 (regression)** や**あてはめ (fitting)** [Bishop 06, Bishop 08] という機械学習や統計的予測の問題に帰着させ解く。

まず、回帰モデルを、評価値行列 \mathbf{R} 行列の分解として捉えてみよう。与えられた行列 \mathbf{R} の未評価アイテムの部分は欠損しているが、欠損していない完全な評価値行列 \mathbf{R}^* を考え、この行列を次のように分解する。

$$\mathbf{R}^* \approx \mathbf{U}^T \mathbf{V} \quad (10.2)$$

ここで、 \mathbf{U} と \mathbf{V} の大きさは、それぞれ $K \times n$ と $K \times m$ である。 \mathbf{U} の第 x 列ベクトル \mathbf{u}_x は利用者 x の特徴を、 \mathbf{U} の第 y 列ベクトル \mathbf{u}_y はアイテム y の特徴を表している。すると、利用者 x へのアイテム y への評価値は $r_{xy}^* \approx \mathbf{u}_x^T \mathbf{v}_y$ のようなモデルで表される。 \mathbf{u}_x の要素を説明変数と、 \mathbf{v}_y の要素をパラメータとみなせば式 (10.1) と同等のモデルであり、逆に \mathbf{v}_y を説明変数にすれば、式 (9.2) と類似したモデルとも解釈できる。ここでもし K が $\max\{m, n\}$ なら、近似ではなく厳密に $\mathbf{R}^* = \mathbf{U}^T \mathbf{V}$ となるように分解できる。だがこれでは、観測データを書き写しただけで、嗜好パターンを要約したモデルとはいえない。また、実際に観測できる評価値行列は \mathbf{R}^* ではなく、欠損のある \mathbf{R} である。そこで、 K を $K \ll m, n$ に固定し、 \mathbf{R}^* との期待的な損失を最小化するよう

に、 \mathbf{R} を \mathbf{U} と \mathbf{V} に分解することで、モデルを獲得する。

こうした分解をするには、**行列分解** (matrix decomposition) の手法を用いる。こうした手法の一つ [Canny 02b] を述べる。この方法では、残差 $(\mathbf{R} - \mathbf{U}^T \mathbf{V})$ の各要素が正規分布に従うとしてモデル化し、観測された評価値の生成確率が高くなるように \mathbf{U} と \mathbf{V} を計算する。この行列分解手法を**因子分析** (factor analysis) という。分解ができれば、任意の利用者の任意アイテムに対する推定評価値 $\hat{r}_{xy} = \mathbf{u}_x^T \mathbf{v}_y$ が計算できる。しかし、実際の \mathbf{R} には欠損値があるので、文献 [Canny 02b] では、これらの欠損値を潜在変数とみなして EM アルゴリズム [Dempster 77] を適用して解いている。欠損値に対処する方法は、潜在変数以外にも、全員が必ず評価するアイテム集合を利用する、欠損値を平均値などで補完する、欠損した要素は無視して損失を計算する [Bell 07b] といった方法もある。回帰モデルは単純なので、演算操作が複雑になるプライバシー保護協調フィルタリング (14 章) などへの適用や、カーネルを使った非線形回帰を導入して予測の向上を図るといった拡張が可能である。

次に文献 [Aggarwal 99] の **Horting** 法を紹介する。式 (10.1) のように全体を一つの線形モデルで表すと、大まかすぎて予測精度が低くなる場合がある。そこで、嗜好が類似している利用者の中で局所的に線形モデルを構築する。利用者 a について、共通に評価したアイテムが十分に多い利用者 x を見つける。さらに、これらの利用者 x の評価値 r_{xy} から $\hat{r}_{ay} = \theta_{ax} r_{xy} + b_{ax}$ の線形関数で、利用者 a のスコアを高精度で予測できる利用者を選び出す。このとき、利用者 x は利用者 a を予測可能とイいう。なお、 θ_{ax} や b_{ax} は、両者が共通に評価しているアイテムの評価値から計算可能である。次に、利用者をノードとし、 x から a への予測可能性を逆向きの a から x への有向辺で示すグラフを生成しておく。このグラフを用いて、利用者 a のアイテム y への評価値を推定する。利用者 a から直接リンクしている利用者で、アイテム y を評価している利用者の集合を \mathcal{X}'_y とする。この集合が空でなければ、アイテム y への予測評価

値は、 \mathcal{X}'_y 中の利用者それぞれによる予測評価値の平均とする。

$$\hat{r}_{ay} = \frac{1}{|\mathcal{X}'_y|} \sum_{x \in \mathcal{X}'_y} \theta_{ax} r_{xy} + b_{ax}$$

もし、 \mathcal{X}'_y が空なら、リンクを 2 段たどり、アイテム y を評価している利用者集合を求め \mathcal{X}''_y とする。この集合が空でなければ次式で予測評価値を求める。

$$\hat{r}_{ay} = \frac{1}{|\mathcal{X}''_y|} \sum_{x \in \mathcal{X}''_y} \theta_{ax'} (\theta_{x'x} r_{xy} + b_{x'x}) + b_{ax'}$$

ただし、 x' は、各 x について x と a を中継する利用者である。さらに \mathcal{X}''_y も空ならば、3 段以上のリンクを考慮する。

10.2.2 クラス分類問題に帰着させる方法

回帰と並ぶ代表的な機械学習の枠組みである**クラス分類** (classification) も利用できる。アイテム y への評価値 r_{ax} は、 \mathcal{R} 中の値の一つをとるので、その大小関係を見捨てる。アイテム y への評価値 r_{ax} は、 \mathcal{R} 中の値の一つをとるので、その大小関係を見捨てる。また、分類するアイテム y の特徴量には、このアイテムへの他の利用者による評価値や、活動利用者の他のアイテムへの評価値が利用できる。あとは、これらの特徴量で表されたアイテムが分類されるべきクラス、すなわち評価値を予測するモデルをクラス分類の学習手法によって獲得すればよい。

このような方法として、**逐次型二項関係学習法** (Cross-G-Learn-Relation method) [Nakamura 98, 山西 03] を紹介する。なお、原論文では評価大小関係を考慮する工夫がさらになされているが、ここでは簡潔に記す。これは、各クラスごとに、対象の分類されやすさを表す関数を獲得し、その関数の値が最大になるクラスに対象を分類する。利用者 x のアイテム y の評価値を予測するとき、逐次型二項関係学習法では、アイテム y への他の利用者の評価と、利用者 x のその他のアイテムへの評価を入力とし

た次の関数を用いる

$$\hat{r}_{xy} = \arg \max_{r \in \mathcal{R}} \left(\sum_{i.s.t. r_{iy}=r} v_{xi} + \sum_{j.s.t. r_{xj}=r} w_{yj} \right) \quad (10.3)$$

利用者 x が仮にアイテム y の評価値を r としたとき、第 1 項ではアイテム y についての評価が同じ r である利用者 i について、利用者間重み v_{xi} の総和を求める。同様に、第 2 項では利用者 x が同じ r と評価をしているアイテム j について、アイテム間重み w_{yj} の総和を求める。すなわち、類似利用者と類似アイテムの評価値の両方を考慮している。重み v_{xi} と w_{yj} は **オンライン学習** (online learning) [丸岡 99] の枠組みで求める。これは、行列 \mathbf{R} の形式で評価値がまとめて与えられてから学習するのではなく、誰かが何かアイテムを評価して、 r_{xy} が一つ与えられるごとに重みをより適切なものに更新する方法である。各評価値を与えられるごとに、予測値とその与えられた値とを比較することを繰り返し、ある時点までの累積の誤りを小さくするようにモデルを学習する。もう少し詳しく述べると、最初に全ての重み v_{xi} と w_{yj} を初期化する。その後、 r_{xy} が観測されるたびに、利用者 x 以外でアイテム y を評価済みの全ての利用者 i について、 $r_{xy} = r_{iy}$ なら重み v_{xi} を増やし、そうでないなら減らす。さらに、利用者 x が評価している y 以外の全てのアイテム j について、評価値が一致すればやはり重み w_{yj} を増やし、そうでなければ減らす。このように評価値が与えられるごとに重みを更新する。この方法はオンライン学習なので、モデルベースだが、アイテムや利用者が追加されても対応できる特徴がある。ただし、アイテムや利用者の削除には対応できない。

10.2.3 順序回帰問題に帰着させる方法

5.2 節では、採点法や格付け法による評価値は、本来は順序付カテゴリとして扱うべきものであることを述べた。すなわち、評価値が 2 のアイテムは、4 のアイテムの半分程度好きなのではなく、評価値 3 のアイテムより、4 のアイテムの方がより好きであるということだけを示している。こうした順序付カテゴリ値を予測する問題は順序回帰

と呼ばれる。この順序回帰を、ブースティング (boosting) [Freund 96, フロイド 99] の枠組みで扱う **RankBoost** [Freund 98, Freund 03] で推薦をする研究などがある。後に、この順序回帰を用いた、内容ベースと協調フィルタリングのハイブリッド法を 13.1 節で紹介する。

10.3 確率モデル

モデル化に用いた関数が確率分布として解釈できるものを本稿では特に確率モデルと呼ぶ。これら確率モデルを、大きく履歴条件型と共起型の二つに分け、順に説明する。

10.3.1 履歴条件型

履歴条件型の確率モデルでは、活動利用者のアイテム j への評価値を次の条件付期待値で予測する [Breese 98].

$$\begin{aligned}\hat{r}_{ay} &= \mathbb{E}[r_{ay} | r_{aj} \text{ s.t. } j \in \mathcal{Y}_a] \\ &= \sum_{r \in \mathcal{R}} r \Pr[r_{ay} = r | r_{aj} \text{ s.t. } j \in \mathcal{Y}_a]\end{aligned}\tag{10.4}$$

これは、活動利用者の過去の嗜好データが与えられたときの、活動利用者のアイテム y への評価値の条件付期待値である。実際には、式 (10.4) 中の確率分布は未知なので、推定した関数を利用する。式中では他の標本利用者の嗜好データが参照されていないが、確率分布の推定の過程で、これらの嗜好データを利用するため、協調フィルタリングによる推薦といえる。また、適合/不適合の 2 値の嗜好データ、すなわち $\mathcal{R} = \{0, 1\}$ の場合、式 (10.4) は単なる条件付確率分布となる。

$$\hat{r}_{ay} = \Pr[r_{ay} = 1 | s_{aj} \text{ s.t. } j \in \mathcal{Y}_a]\tag{10.5}$$

このように簡潔になるので、予測評価値を提示する評価閲覧よりも、適合か不適合かの判定ができればよい適合アイテム発見タスクに適す。この履歴条件型モデルでは、

条件付確率 (10.5) の計算のため、肯定的評価の $r_{xy} = 1$ と否定的評価の $r_{xy} = 0$ の両方の場合の訓練事例が必要である。しかし、5 章で述べたように、暗黙的な評価では、購入や閲覧などの行動がなかったことで否定的な評価とみなす。すると、未評価と否定的評価は混同され、 $r_{xy} = 0$ となる事例にはノイズが多くなる。そのため、未評価と否定的評価が明確に区別される事例が必要となり、この履歴条件型ではこうした暗黙的な評価では不利になる。

式 (10.4) 中の条件付確率は、 \mathcal{Y}_a に依存するため、各利用者について個々にモデルを獲得する必要が生じ、実用的ではない。そこで実際には、利用者には依存しない、次の全アイテムへの評価の同時分布を求める。

$$\Pr[r_{\cdot y} \text{ s.t. } y \in \mathcal{Y}] \quad (10.6)$$

そして、式 (10.4) 中の条件付分布は、ベイズ則と不要な変数を周辺化で消去することで計算する。だが、各 $r_{\cdot y}$ は $|\mathcal{R}|$ 個の値をとることが可能で $|\mathcal{Y}| = m$ なので、この分布の定義域は $\mathcal{R}^{\mathcal{Y}} = \mathcal{R}_1 \times \mathcal{R}_2 \times \dots \times \mathcal{R}_m$ である。よって、この分布の飽和モデルのパラメータ数は $|\mathcal{R}|^m - 1$ となり、 m に対して指数的に増加する。しかし、データは疎で m は大きいので、単純に評価値の各組み合わせの頻度を数え上げて、この飽和モデルのパラメータを推定することは現実的ではない。そこで、このような場合の一般的な対策を導入し、変数 $r_{\cdot y}$ の間に部分的な条件付独立性を仮定し学習すべきパラメータ数を減らす。このような条件付独立性を導入した確率モデルを、依存関係をグラフで表現して示すため、一般にグラフィカルモデルという。その一つである**ベイジアンネットワーク (Bayesian network)** [繁榘 06] は [Breese 98] で利用されている。ベイジアンネットワーク全般の問題として独立性の構造をうまく設計する難しさがあるが、うまく設計できれば学習も、推薦も効率的に実行できる。また、式 (10.4) の条件付確率を直接内部構造としてもつ確率モデルである**依存性ネットワーク (dependency network)** を使う [Heckerman 00] などもある。このモデルでは、同時確率から条件付確率へ変換が不要なので、実行時の推薦は高速に実行できるが、事前のモデル学習の計算量は増える。

10.3.2 共起型

もう一方の共起型の確率モデルについて述べる。このモデルでは、ある利用者が、あるアイテムを評価したことを、その利用者が x であるという事象と、そのアイテムが y であるという事象が共起しているととらえる。そして、この共起確率をモデル化する。このモデル化には、自然言語処理で文書と語の共起確率を表すために考案された**確率的潜在意味解析 (probabilistic latent semantic analysis; pLSA)** [Hofmann 99a] を利用する。この確率モデルは非常に柔軟で、純粋な協調フィルタリングだけでなく、評価の揺らぎ、アイテムの情報、コンテキスト情報、および個人属性情報などの要因を導入するといった拡張が可能である。

本節では、利用者の嗜好データのみに基づく協調フィルタリングを対象としたモデル [Hofmann 99c] を紹介する。最初に、利用者とアイテムの共起関係だけを使うモデルについて述べる。利用者とアイテムに対し、それぞれ $\mathcal{X} = \{1, \dots, n\}$ と $\mathcal{Y} = \{1, \dots, m\}$ 中のいずれかの値を取る多値の確率変数 X と Y を割り当てる。嗜好データを、利用者 x がアイテム y を購入するという行為で暗黙的に獲得する場合を考える。これは、 $X = x$ という事象と、 $Y = y$ という事象が共起していることに相当する。また、 $\mathcal{R} = \{0, 1\}$ であるとき、 $r_{xy} = 1$ となる対 (x, y) が観測されることとも等価である。履歴条件型の確率モデルでは $r_{xy} = 0$ となる事例も必要であったが、共起型では $r_{xy} = 1$ となる事例だけでモデルが学習できる。このため、未評価と不支持が区別できない暗黙的評価の場合に、この共起型確率モデルは便利である。

さらに、このモデルで重要な役割をはたす**潜在変数 (latent variable) Z (隠れ変数 (hidden variable))** ともいう) を導入する。これも多値変数で $\mathcal{Z} = \{1, \dots, K\}$ 中の値をとり、潜在的な嗜好のパターンを表す。この潜在変数の値はデータとして観測されることはなく、モデルの中でのみその存在を仮定する。これはクラスタモデルと類似しているので、灰色の羊問題 (10.1 節) が生じるように思える。だが、 K 個のパターンのうちどれか一つに割り当ててのではなく、それらが混ざり合った状態を考慮する

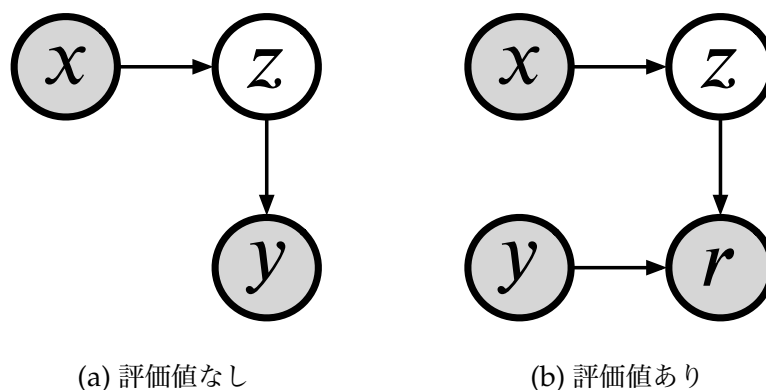


図 10.1: pLSA モデル [Hofmann 99c]

ので、この問題は生じない。そして、この潜在変数を用いて、 x と y の同時確率を、次式で表すのが pLSA モデルである。

$$\Pr[x, y] = \sum_{z \in \mathcal{Z}} \Pr[x] \Pr[z|x] \Pr[y|z] \quad (10.7)$$

$$= \sum_{z \in \mathcal{Z}} \Pr[x|z] \Pr[y|z] \Pr[z] \quad (10.8)$$

このモデルでは、 z が与えられたときに x と y が条件付独立であることを仮定して、モデルのパラメータの総数を減らしている。また、 $\Pr[x|z]$ 、 $\Pr[y|z]$ 、 $\Pr[z]$ はそれぞれカテゴリ分布に従う。

図 10.1(a) は、**グラフィカルモデル (graphical model)** という表現方法で pLSA モデルを図示したものである。このグラフで、灰色のノードは値が観測される X や Y のような**観測変数 (observed variable)** (**可視変数 (visible variable)** ともいう) を、白いノードはその値が観測されない潜在変数を表す。有向辺は確率的な依存関係を表し、辺が出ている変数に、辺が入っている変数が依存することを表す。また、この図にはないが、値が確定しているパラメータは黒い点で表し、同じ確率変数を複数回サンプリングすることを示すプレートという表現もある。これらの表現については教科書 [Bishop 08, 8 章]などを参考にされたい。

モデルのパラメータは $\Theta = (\{\Pr[x|z]\}, \{\Pr[y|z]\}, \{\Pr[z]\})$ であるが、これらを最尤

推定で求める。すなわち、いろいろな利用者がいろいろなアイテムを評価した、 N 個のデータ $D = \{(x_i, y_i)\}_{i=1}^N$ に対する次の対数尤度を最大にするように求める。

$$\Pr[D; \Theta] = \sum_{(x,y) \in D} \ln \Pr[X = x, Y = y; \Theta]$$

潜在変数があるため、この最尤推定は **EM アルゴリズム** (EM algorithm) [Dempster 77, Bishop 08] によって行う。具体的には次の二つのステップを交互に反復する。一つ目の E ステップでは、パラメータ $\{\Pr[z|x]\}, \{\Pr[y|z]\}, \{\Pr[z]\}$ が与えられたときの、潜在変数の分布を求める。

$$\Pr[z|x, y] = \frac{\Pr[z] \Pr[x|z] \Pr[y|z]}{\sum_{z'} \Pr[z'] \Pr[x|z'] \Pr[y|z']} \quad (10.9)$$

二つ目の M ステップでは、前のステップで求めた分布 $\Pr[z|x, y]$ を用いてパラメータを更新する。

$$\Pr[x|z] = \frac{\sum_y n(x, y) \Pr[z|x, y]}{\sum_{x', y} n(x', y) \Pr[z|x', y]} \quad (10.10)$$

$$\Pr[y|z] = \frac{\sum_x n(x, y) \Pr[z|x, y]}{\sum_{x, y'} n(x, y') \Pr[z|x, y']} \quad (10.11)$$

$$\Pr[z] = \frac{\sum_{x, y} n(x, y) \Pr[z|x, y]}{N} \quad (10.12)$$

ただし、 $n(x, y)$ は、 x と y が共起している D 中の対の数である。以上の手続きを収束するまで反復すると、パラメータが計算できる。そして、次式の $\Pr[y|x]$ を求めておき、

$$\Pr[y|x] = \frac{\sum_z \Pr[x|z] \Pr[y|z] \Pr[z]}{\sum_{y', z} \Pr[x|z] \Pr[y'|z] \Pr[z]} \quad (10.13)$$

活動利用者 a がアイテム y を好む度合いは $\Pr[Y = y|X = a]$ の大きさを測る。活動利用者 a に対しては、この度合いを最大化する次のアイテム y^* を推薦すればよい。

$$y^* = \arg \max_{y \in \mathcal{Y} \setminus \mathcal{Y}_a} \Pr[Y = y|X = a] \quad (10.14)$$

なお pLSA では訓練データ中に現れない利用者，すなわち， $a \notin \mathcal{X}$ の場合にはこの方法では推薦できない．このような新規利用者を扱うことを情報検索の文脈では folding-in という．この folding-in を行う方法として，潜在変数とアイテムしか現れないパラメータはそのまま固定し，利用者が関連するパラメータ $\Pr[x|z]$ のみを更新する手続きが提案されている [Hofmann 99b]．

次に，利用者 X とアイテム Y に，評価値を表す確率変数 R も加えた拡張を考える．この変数は，評価値の値域 \mathcal{R} 中の値をとる多値変数である．このモデル化では $X = x$ ， $Y = y$ ，および $R = r$ の共起確率 $\Pr[x, y, r]$ を扱う．そして，利用者 a のアイテム y の推定評価値 \hat{r}_{xy} は最頻値

$$\hat{r}_{xy} = \arg \max_{r \in \mathcal{R}} \Pr[R = r | X = a, Y = y]$$

や期待値

$$\hat{r}_{xy} = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} r \Pr[R = r | X = a, Y = y]$$

で計算する．なお， $\Pr[r|x, y]$ は $\Pr[x, y, r]$ から計算できる．図 10.1(b) は $\Pr[x, y, r]$ のモデル化の一例である．このモデルでは，利用者 x は z で表されるグループに分類され，そのグループ z とアイテム y に依存して評価値 r が決まると解釈できる．モデルを決めれば，あとは対数尤度を最大化するパラメータを EM アルゴリズムで求めればよい．文献 [Hofmann 99c] では他にもいくつかのモデルを挙げている．これらのモデルは評価値を離散変数で表し，カテゴリ分布に従うと考えているので，評価値の大小関係を見捨てている．そこで離散の評価値を実数値として扱うことで大小関係を考慮し，それがガウス分布に従うとするモデルもある [Hofmann 03]．どのモデルが適切かだが，一般には 3.1 節で述べた交差確認法で，予測精度を最大にするようなものを選ぶ．

この共起型の確率モデルは自由に設計できる余地が多く，いろいろな拡張が可能である．5.2 節では，利用者の評価値に一貫性はなく揺らぎがあると述べた．こうした揺らぎを考慮する 2 種類の方法を，文献 [Jin 03] では提案している．図 10.2(a) の

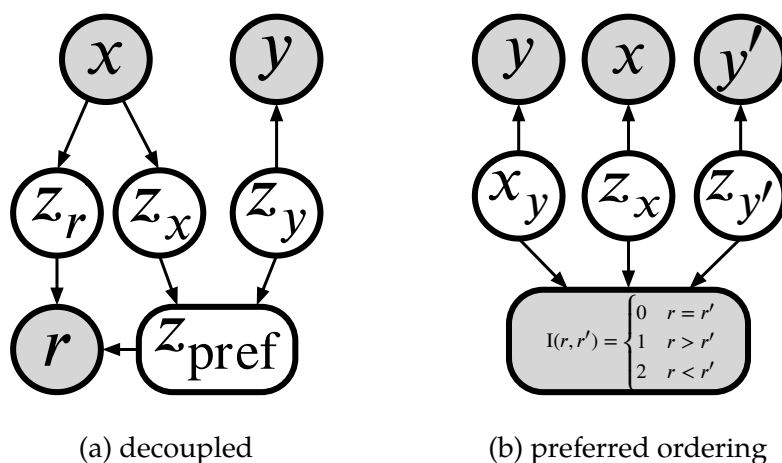


図 10.2: 評価値の揺らぎを考慮したモデル [Jin 03]

decoupled は、まず揺らぎのない真の評価値は分からないので、これを潜在変数 z_{pref} で表す。この真の評価値は、同じ嗜好の利用者のグループを示す z_x と、類似したアイテムのグループを表す z_y に依存して決まる。さらに、評価値がどのように揺らぐのかというパターンに利用者は分けられ、そのパターンを潜在変数 z_r で表す。観測される評価値 r は真の嗜好 z_{pref} と揺らぎのパターン z_r に依存して決まる。推薦は、予測した真の評価値 z_{pref} によって行う。もう一つの preferred ordering モデル（図 10.2(b)）は、評価値を順序尺度と考える方法である。同じ利用者 x が二つのアイテム y と y' のそれぞれに r と r' の評価値を与えた場合を考える。このとき、 r と r' の順序関係だけを取り出す関数 $I(r, r')$ を導入し、この関数 I の値が、利用者アイテムのグループを表す潜在変数 z_x と z_y に依存するというモデル化である。このようにモデルを決めれば、あとは EM アルゴリズムでパラメータを学習できる。これらのモデルでは揺らぎを扱うことができるが、パラメータの総数は増えるので、それに応じた十分な嗜好データが必要になる。

pLSA では新規の利用者に対応できない問題があったが、これにベイズの枠組に拡張することで対応する **潜在的ディリクレ配分法 (latent Dirichlet allocation; LDA)** [Blei 03] がある。式 (10.7) の pLSA モデルでは、最初に既知の利用者を \mathcal{X} の中から $\Pr[x]$ に従って選択し、その利用者の嗜好パターンの分布 $\Pr[z|x]$ に基づいて選ん

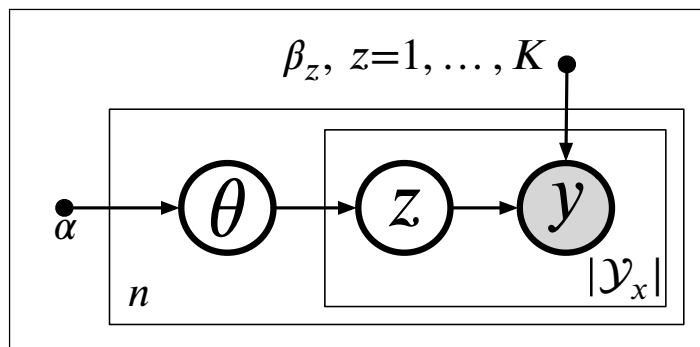


図 10.3: 潜在的ディリクレ配分法 (LDA) のモデル [Blei 03]

だ嗜好パターンに基づいてアイテムを選択する。そのため、訓練データに表れない利用者については、その嗜好パターンが不明であり、pLSA では対応できない。そこで、嗜好パターン θ 自体をディリクレ分布に従って生成するように拡張することで、新規利用者に対応できるようにしたのがこの LDA である。この LDA では、訓練データ中の利用者 x は、しかし、変分ベイズや MCMC を用いた近似計算が必要になり、pLSA より計算量は増大する。この LDA と類似したモデルとして [Marlin 04] などもある。

10.4 時系列モデル

時間に伴う変化を嗜好の予測に利用する方法を紹介する。5.4 節で述べたように、標本利用者の時間順の行動や評価の履歴情報を、利用者 DB に蓄積している場合がある。この情報から、利用者の行動の時間的推移を表すモデルを構築し、活動利用者の現在にいたる行動や評価とこのモデルから、嗜好を予測する手法を紹介する。

10.4.1 基本的な時系列モデル

レンタルビデオ店で、先々週はあるドラマの第 1 話を、先週はこのドラマの第 2 話を借りていった顧客がいたとする。このとき、今週はこのドラマの第 3 話を借りるだろうことは容易に予測できる。この購入履歴に基づくモデルを利用した文献 [Pavlov 03] の方法を紹介する。利用者が t 回目に購入したアイテムを $y^{(t)}$ で表すと、購入履歴の

系列は $\langle Y^{(t)} \rangle = \langle y^{(1)}, y^{(2)}, \dots, y^{(t)} \rangle$ となる。例えば、上記の連続ドラマの例であれば $y^{(1)} \sim y^{(3)}$ は、それぞれドラマの第 1~3 回に相当する。この購入履歴が与えられたときの、次に購入するアイテムの条件付確率 $\Pr[y^{(t+1)} | \langle Y^{(t)} \rangle]$ を求めておき、この確率を最大にするアイテムを推薦する手法を考える。これは、標本利用者の購入履歴を数え上げれば原理的には計算可能である。しかし、3.2 節で述べたようにデータが疎であるので、いたるところで確率が 0 となり、実際にはこの方法では計算できない。このような場合には、 K 回前までの状態、すなわち購入アイテムに基づく K 重マルコフモデルを用いることが多い。しかし、推薦の場合にはかなり過去の情報も次の状態に影響するため、単純にこのモデルを適用してもあまり有効ではない。

そこで、自然言語処理の言語モデルでよく利用される **最大エントロピーモデル (maximum entropy model; MaxEnt model)** を導入する。

$$\Pr[y^{(t+1)} | \langle Y^{(t)} \rangle] = \frac{1}{Z} \exp \left[\sum_{k=1}^K \lambda_k f_k(y^{(t+1)}, \langle Y^{(t)} \rangle) \right] \quad (10.15)$$

ただし、 Z は正規化定数、 λ_k は重みパラメータである。ここで **素性関数 (feature function)** $f_k(y^{(t+1)}, \langle Y^{(t)} \rangle)$ が重要になる。素性関数は購入履歴 $\langle Y^{(t)} \rangle$ にある特徴があるとき、次回に購入するアイテムがある特定のものになるなら 1 をとり、それ以外では 0 となるような関数である。例えば、上記のドラマの場合、購入履歴にドラマの第 2 回が含まれていて、次回購入がドラマの第 3 回なら 1 になる関数である。このような素性関数を **trigger** と呼ぶ。この **trigger** には、購入履歴中にアイテム y があるときに、次回にアイテム y' を購入する確率 $\Pr[y^{(t+1)} = y' | y \in \langle Y^{(t)} \rangle]$ と $\Pr[y^{(t+1)} = y']$ との差が大きなものを選ぶ。こうして素性関数を選択すれば、式 (10.15) のモデルのパラメータ λ_k は最大エントロピー原理 [北 99] に基づいて推定できる。素性関数には、**trigger** 以外にも、いろいろなものが利用できる。次に購入するアイテムが、 $s \in 0, 1, 2, \dots$ 個前に購入したアイテムに依存して決まる **gap** マルコフモデルを素性として利用する方法 [岩田 07]。利用者自身の個人属性情報を考慮するのも有効であろう。系列パターンマイニング [Han 11, Pei 01] などと組み合わせれば、**trigger** よりも複雑なパターンも

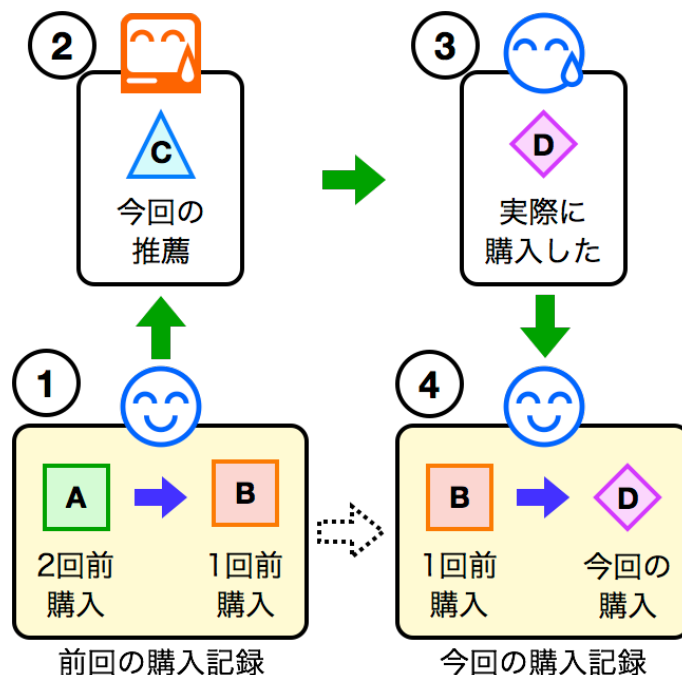


図 10.4: マルコフ決定過程によるモデル

素性関数として利用できるだろう。

10.4.2 マルコフ決定過程モデル

前節のモデルでは、過去の履歴に基づいて、次に利用者が選択する確率が最も高いアイテムを推薦する。だが、推薦がなければそのアイテムを買ったかもしれないが、推薦の影響によってそのアイテムを選ばないという場合もある。加えて、1年間全体で見て、選択したアイテムの価格の総和を最大化するとか、選択したアイテムへの利用者の評価の総和を最大にするような推薦をしたいとする。これらの目的を達成するため、**マルコフ決定過程** (Markov decision process) を使う手法が提案されている [Shani 02, Shani 05].

マルコフ決定過程の前に**マルコフ過程** (Markov process) について述べる。この過程では、次に選択するアイテム $y^{(t+1)}$ は、直前の K 個のアイテム系列 $\langle Y^{(t)} \rangle = \langle y^{(t-K+1)}, \dots, y^{(t)} \rangle$ に依存する。このアイテム系列で状態を表し、状態の遷移確率は

$\text{tr}_{\text{MC}}(\langle Y^{(t)} \rangle, \langle Y^{(t+1)} \rangle)$ と記す。この確率は、十分なデータがあれば、利用者の購買履歴を数え上げれば計算できる。しかし、データは疎なので、文献 [Shani 02, Shani 05] では skipping や、アイテムのクラスタリングなどの手法を用いて対処している。

次にマルコフ決定過程を図 10.4 を用いて説明する。図中①は現在の状態 $\langle Y^{(t)} \rangle = \langle y^{(t-1)} = A, y^{(t)} = B \rangle$ で、この時刻 t にアイテム D を購入して、図中④の次の状態 $\langle Y^{(t+1)} \rangle = \langle y^{(t)} = B, y^{(t+1)} = D \rangle$ に移る。ここで、単に $\langle Y^{(t)} \rangle$ のみに依存して $\langle Y^{(t+1)} \rangle$ が決まる (図の点線矢印) ならマルコフ過程である。それとは異なり、マルコフ決定過程では、状態 $\langle Y^{(t)} \rangle$ で行う推薦という行動 (action) を明示的に考慮する。これにより、推薦という行動によって、利用者の選択が変わることをモデル化できる。この時刻 t での行動 $A^{(t)}$ は図中②にあたり、アイテム C を推薦した。利用者は $\langle Y^{(t)} \rangle$ と $A^{(t)}$ に依存して、アイテム $y^{(t+1)}$ を選択し、次の状態 $\langle Y^{(t+1)} \rangle$ に遷移する。図の例では、推薦を受け入れず利用者は③でアイテム D を購入したので、遷移した状態は④の $\langle Y^{(t+1)} \rangle = \langle B, D \rangle$ となった。この状態遷移を $(\langle Y^{(t)} \rangle, A^{(t)}, \langle Y^{(t+1)} \rangle)$ と、この遷移確率を $\text{Pr}_{\text{MDP}} [\langle Y^{(t)} \rangle, A^{(t)}, \langle Y^{(t+1)} \rangle]$ と記す。また、状態 $\langle Y^{(t)} \rangle$ で行動 $A^{(t)}$ を決定する手続きを方策 (policy) と呼ぶ。さらに、図の③にてアイテム D を選択したが、これに依存して報酬 (reward) が発生する。この報酬は、アイテムへの代金や、利用者のアイテムへの評価値などである。報酬の導入により、個別の推薦ごとの効用を最大化するのではなく、複数の推薦を含む長期間の効用の最大化を考慮できるようになる。この報酬を最大化するような政策を決める学習問題は**強化学習** (reinforcement learning) [Sutton 98, Sutton 00] と呼ばれる。既存の強化学習の方法を用いて、適切な政策が獲得できれば、推薦は政策に従うことで実行可能になる。ただし、データが疎である問題に対処するため、マルコフ決定過程の遷移確率 $\text{Pr}_{\text{MDP}} [\langle Y^{(t)} \rangle, A^{(t)}, \langle Y^{(t+1)} \rangle]$ を、マルコフ過程の遷移確率 $\text{Pr}_{\text{MC}} [\langle Y^{(t)} \rangle, \langle Y^{(t+1)} \rangle]$ によって初期化するなどのヒューリスティックを用いている。

10.4.3 定期購入サービス

時系列を扱う手法の最後に**定期購入** (subscription) というサービスを対象とした手法を紹介する [Iwata 06]. 定期購入サービスとは音楽配信などのサービスの販売形式の一つで、契約すると、解約するまでの期間、任意の曲を任意の回数試聴できる。そのため、個々の購入での満足ではなく、サービス全体への顧客満足度を向上させて、解約を防ぐことが目標になる。上記の強化学習の枠組みでは、各購入行動ごとに時間 t が経過する。一方、サブスクリプションでは、各行動が何回目であることを示す t に加え、実時間と共に推移する契約期間 T もモデル化する必要がある。

利用者 x に t 回目にアイテム $y^{(t)}$ を推薦したとき、現在の契約期間中に解約しない事象 e が生じる確率を最大化するようなアイテムを推薦する。

$$\hat{y}^{(t)} = \arg \max_{y^{(t)} \in \mathcal{Y}} \Pr[e|x, y^{(t)}]$$

推薦の結果、実際に購入したアイテムを $y'^{(t)}$ で表し、この確率を次のように分解する。

$$\Pr[e|x, y^{(t)}] = \sum_{y'^{(t)} \in \mathcal{Y}} \Pr[e|x, y'^{(t)}] \Pr[y'^{(t)}|x, y^{(t)}]$$

利用者 x がアイテム $y'^{(t)}$ を選択したとき解約しない確率が $\Pr[e|x, y'^{(t)}]$ で、 $y^{(t)}$ を推薦したときに $y'^{(t)}$ を選択する確率が $\Pr[y'^{(t)}|x, y^{(t)}]$ である。前者の確率は、**ハザード関数** (hazard function) $h(T|\langle Y^{(t)} \rangle)$ で表す。これは購入履歴が $\langle Y^{(t)} \rangle = \langle y^{(1)}, \dots, y^{(t)} \rangle$ である利用者のうち、期間 T まで契約を継続し、かつ、この時点で契約をやめる割合を示す。この関数は生存時間分析 (survival analysis) の **Cox 比例ハザードモデル** (Cox proportional hazards model) でモデル化する。 $\langle Y^{(t-1)} \rangle$ のあとに $y'^{(t)}$ を購入した系列を $\langle Y^{(t-1)}, y'^{(t)} \rangle$ で表すと $\Pr[e|x, y'^{(t)}]$ は、適当な仮定の下、次式となる。

$$\Pr[e|x, y'^{(t)}] = \frac{h(T|\langle Y^{(t-1)} \rangle)}{h(T|\langle Y^{(t-1)} \rangle) + h(T|\langle Y^{(t-1)}, y'^{(t)} \rangle)}$$

一方の $\Pr[y^{(t)}|x, y^{(t)}$ は、アイテムを推薦することにより選択確率が定数倍されると仮定して、利用者 x の選択確率 $\Pr[y^{(t)}|x]$ から求める。この $\Pr[y^{(t)}|x]$ は、購入履歴に基づく素性関数を用いた最大エントロピーモデルでモデル化する。このようにモデル化すれば、あとは購入履歴からパラメータを学習すればよい。

第 11 章

協調フィルタリング：ハイブリッド

協調フィルタリングはメモリとモデルベースとに分類できると述べたが、これらの中間的な手法も存在する。

personality diagnosis 法 [Pennock 00a] は、メモリベースのように利用者間の類似度を用いる。そして、この類似度を重みとした、各利用者の評価値の重み付平均を推定評価値とする。これだけであれば、メモリベース法だが、類似度計算にモデルベースの要素がある。各利用者個人モデルを作成し、活動利用者と標本利用者の個人モデルが一致する確率を類似度としている。

クラスタリングをしたあとに、メモリベース法を適用する方法は、メモリベース法に、モデルベースの要素を加えた手法と解釈できる。クラスタリングする対象が、利用者集合のもの [Xue 05] と、アイテム集合のもの [O'Connor 99] とがある。前者の方法では、アイテムへの嗜好パターンが類似した利用者のクラスタを生成する。活動利用者と嗜好パターンが近いクラスタを見つけ、そのクラスタ内の利用者を対象に利用者間型メモリベース法を適用する。利用者間型メモリベース法で近傍利用者を利用した場合と同様に、推薦の高速化と、場合によっては予測精度の向上が見込めるが、適応性の面では不利になる。後者のアイテム集合をクラスタリングする方法では、9.2

節のように、評価値行列 \mathbf{R} の列ベクトルの類似性に基づいてクラスタを作る。その後、各クラスタごとに個別にメモリベース手法を適用する。これは、特定のアイテムカテゴリ内に推薦対象を限定すると、そのアイテム群に特に関心のある利用者に対する予測精度が向上するとの考えに基づくが、実験ではその有効性は確認できなかつたと報告している。

第 12 章

内容ベースフィルタリング

6 章で述べたように、内容ベースフィルタリングを、やや広義にとらえ、アイテムや利用者自身の特徴に基づく推薦手法とした。この内容ベースフィルタリングについて、次の三つの観点から述べる。

特徴の種類 アイテムの特徴、個人属性特徴、コンテキストの特徴

入力の形式 嗜好データと検索質問

推薦規則の獲得 学習による獲得と人手による定義

これらのうち、特徴の種類については 5.5 節で述べたので、残り二つについて述べる。

12.1 入力の形式

入力の形式には嗜好データと検索質問とがある。嗜好データは、利用者の好の度合いを示したデータで、間接指定型内容ベースフィルタリング [Condliff 99, 村上 05, 小野 06] で用いられる。個々のデータの推薦への寄与は小さいので、継続的にデータを集積する必要がある。そのため、永続的個人化 (2.1 節) のために主に利用する。

もう一方の検索質問とは、6 章の冒頭で述べたような、アイテムの特徴に対する制約であり、直接指定型の内容ベース [Smyth 04, Smyth 03, Thompson 04, Ricci 07] で利

用される。特に、候補アイテムがテキストで、検索質問に索引語が利用される場合には情報検索 [徳永 99] そのものとなる。こうしたシステムは対話的に動作し、徐々に制約を強くして候補を絞り込むものが多い (16.1 節)。このような明示的な検索質問の他に、暗黙的なものもある。例えば、論文の推薦システムである CiteSeer [Lawrence 99] では、現在閲覧している論文を検索質問とみなし、その論文と本文テキストや引用文献リストが類似している文献を推薦する。検索質問は、継続的に蓄積しなくても利用者の嗜好に関する情報を得られるので、一時的個人化 (2.1 節) のために主に利用する。

12.2 推薦規則の獲得

推薦するアイテムを決定する規則は、機械学習によって獲得する場合と、人手によって定義する場合がある。学習による獲得は、永続的な推薦を嗜好データを用いて行う場合に採用する場合が多い。機械学習問題としては、**クラス分類** (classification) 問題 [Bishop 06, Bishop 08, 麻生 06] に該当する問題である。嗜好データの好き・嫌いをクラスとみなす。一方、特徴ベクトルは、嗜好データを獲得したときのアイテム、個人属性、コンテキストの特徴で構成する。これらのクラスと特徴ベクトルの対を訓練事例とし、機械学習アルゴリズムを適用すれば、アイテム、個人属性、コンテキストの特徴から、利用者のアイテムへの嗜好を予測できるようになる。嗜好データを順序付カテゴリとみなす場合は、順序回帰問題 [Agresti 03] としてとらえることもできる。検索質問入力を採用した場合にも機械学習によって推薦規則を洗練する方法もある [Thompson 04]。この方法では、検索質問で頻繁に利用される特徴や属性値について、類似度判定の際の重みを増やして、将来の検索で重視するようにしている。

もう一方の人手による定義とは、利用者が与えた効用関数、IF-THEN 型ルール、または類似度関数などに基づいて、推薦するアイテムを決める方法である。文献 [Burke 02] では、効用ベース (utility-based) と知識ベース (knowledge-based) とに分類されている方法である。入力が検索質問の場合には、類似度関数を用いて検索質問の近傍を推薦する手法が多用されされている。こうした規則は、人間がヒューリス

ティックに作成するため、複雑で予測精度の高いものを作るのは難しい。だが、純粋に手作業で定義するのではなく、学習により獲得した評価値をヒューリスティックな規則で修正することは比較的容易で、有用である。例えば、特売品の予測評価値を増加させたり、在庫がない商品の予測評価値を下げたりといったように、システム管理者側の意図を反映させる場合などである。

第 13 章

嗜好の予測：まとめ

ここでは、協調フィルタリングと内容ベースの二つの予測手法を組み合わせたハイブリッド法について述べたあと、アルゴリズムの選択の指針について述べる。

13.1 ハイブリッド法

6 章では、協調フィルタリングと内容ベースフィルタリングの 2 種類の推薦手法を紹介し、それぞれの長所と短所を述べた。簡単にまとめると、協調フィルタリングは他の利用者の意見に基づいて推薦する方法で、アイテム特徴のデータベースが不要であることや、多様性の高い推薦ができるなどの利点がある。一方、内容ベースフィルタリングは、アイテムや利用者の特徴に基づいて推薦する方法で、新規のアイテムでも推薦できるなどの利点がある。ここで、他の利用者の意見とアイテムの特徴は相反する情報ではなく、同時に獲得できる。そこで、互いの長所を生かすように、二つの手法を組み合わせたハイブリッド法について述べる。このハイブリッド法の分類を、文献 [Burke 02] の分類に、最近の研究含めて再編した分類を示す。なお、結合が疎なものから、より密接なものへ順に並べた。研究の動向も、より密接な結合方法へ展開しているといえるだろう。

13.1.1 混合 (mixed)

これは、協調と内容ベースの両方の手法による推薦結果を同時に混合させて提示する方法である。どちらを採用するかは利用者に任されている。

論文の適切な査読者を探すシステム [Basu 01] は、内容ベースと協調フィルタリングの二通りの推薦機能を備える。どちらを使うかは、利用者が決める。

13.1.2 切り替え (switching)

何らかの規準に基づいて、内容ベースと協調の推薦手法を切り替える方法である。嗜好データが少ないときは、比較的、疎なデータに強い協調フィルタリングを利用し、十分なデータが蓄積されれば内容ベースに切り替えるという規準がある。また、新規アイテムに対しては、協調フィルタリングは利用できないので、内容ベースで推薦を行うが、そのアイテムへの嗜好データが蓄積されれば協調フィルタリングを使うという規準も考えられる。

13.1.3 メタ推薦 (meta-recommendation)

メタ推薦とは個別に得られた各推薦手法の推薦結果を統合する方法である。情報検索で、複数の検索結果を統合するメタ検索エンジン [Dwork 01] と同様の手法である。この方法には、各手法を改造する必要がないため実装が容易である利点がある。

P-Tango[Claypool 99] では、内容ベースと協調のスコアの重み付線形和を、全体のスコアとする。内容ベースと協調のスコアの重みは初期的には同じである。その後、推薦結果を利用者が受け入れたかどうかのフィードバックに基づいて、それぞれの手法の重みを変化させている。

Pazzani の方法 [Pazzani 99] では、各手法による推薦リストの 1~5 位に、それぞれ 5~1 点を与え、各アイテムの総点数に基づいて最終的な推薦順位を決定する Borda count 法 [Borda 95] によって統合する。

TiVo[Ali 04] では、基本的には予測評価値で整列してアイテムを提示するが、評価値が同じなら協調フィルタリングの方を内容ベースより上位に表示する。

13.1.4 縦続 (cascade)

この手法では、最初の段階で候補集合を生成し、次段階でその候補集合から詳細な推薦をする。候補を限定することで、推薦を高速化し、また、条件から大きくかけ離れたアイテムを推薦することを回避できる。

EntreeC[Burke 02] はレストランの推薦システムである。最初の段階では、価格帯や和洋中など、希望するレストランの特徴を示した検索質問を用いて、内容ベースの手法で候補となるレストランを選び出す。その後、協調フィルタリングによって、候補の順位付けをする。

Google News[Das 07] では、利用者が指定した言語やジャンルや、記事の新しさなどの候補記事を絞り込んだのち、協調フィルタリングでより個人化した推薦を行う。逆に、推薦リストを提示した後に、具体的な条件でフィルタリングする手法 [Swearingen 01] も論じられている。こうした簡単な条件による推薦候補の絞り込みは、実装が容易な割に利用者の満足を高めるとされているので [Swearingen 01]、ぜひ導入しておくべきであろう。

13.1.5 特徴拡張 (feature augmentation)

特徴拡張とは、一方の手法が出力する評価値や分類結果を、もう一方の手法の入力とするアプローチである。上記の縦続接続では、二つの推薦器の出力を優先度をつけるが、特徴拡張では、一方の出力がもう一方の入力となる点が異なる。

Libra[Mooney 99] は書籍の推薦システムで、Amazon.com から取得した書籍情報の特徴に基づいて内容ベースの推薦を実行する。このとき、Amazon.com 内の協調フィルタリングによって抽出された関連書籍を、書籍の特徴に含めることで協調フィルタリングの要素を加えている。協調フィルタリングにより求めた類似アイテムを、

アイテムの特徴に反映させる手法は SmartPad[Lawrence 01] にも見られる。

Good らの方法 [Good 99] は、人間の標本利用者の他に、アイテムの特徴に基づいて嗜好を判断する仮想エージェント利用者も参加させて協調フィルタリングをする。仮想エージェント利用者には、特定のジャンルの映画に高い評価値を与えるものなどがある。こうして、内容ベースの推薦の要素を加える。

13.1.6 抽象情報 (abstraction)

これは、他の利用者の嗜好データや各種の内容データを、ベクトルや確率分布の形式の、メタレベルの抽象的な情報に変換し、別の推薦器に入力する方法である。特徴拡張とは、推薦結果そのものを受け渡さない点が異なる。

Fab[Balabanović 97] は、各利用者ごとに、内容ベースの手法で利用者プロフィール、すなわち、利用者が好むアイテムの特徴を生成する。利用者の類似性をこの利用者プロフィールの類似性で測り、類似した利用者が好むアイテムを活動利用者に推薦する。同様の手法は [Pazzani 99] でも利用されている。

Lee の方法 [Lee 01] は、10.1 節のクラスタモデルの協調フィルタリングで、各クラスタ内のモデルを内容ベースで生成する。

文献 [Yu 03] では階層ベイズに基づく手法が提案されている。活動利用者の嗜好はアイテムの特徴ベクトル \mathbf{f} が与えられたときの評価値の事後分布 $\Pr[r_{ay}|\mathbf{f}_y, \Theta]$ で予測する。すなわち、内容ベースの枠組みに基づく推薦である。しかし、ここではパラメータ Θ の事前分布に無情報事前分布を用いないことでハイブリッド化をする。この事前分布には、利用者 DB の嗜好に依存した分布 $\Pr[\Theta|\mathbf{R}]$ 、すなわち、他の利用者の嗜好データに基づく協調フィルタリングの要素をもつものを用いる。

13.1.7 完全結合 (total integration)

完全結合では、結合される手法に前後関係がなく、他の利用者の情報と内容情報とを同等のレベルでモデル化する。

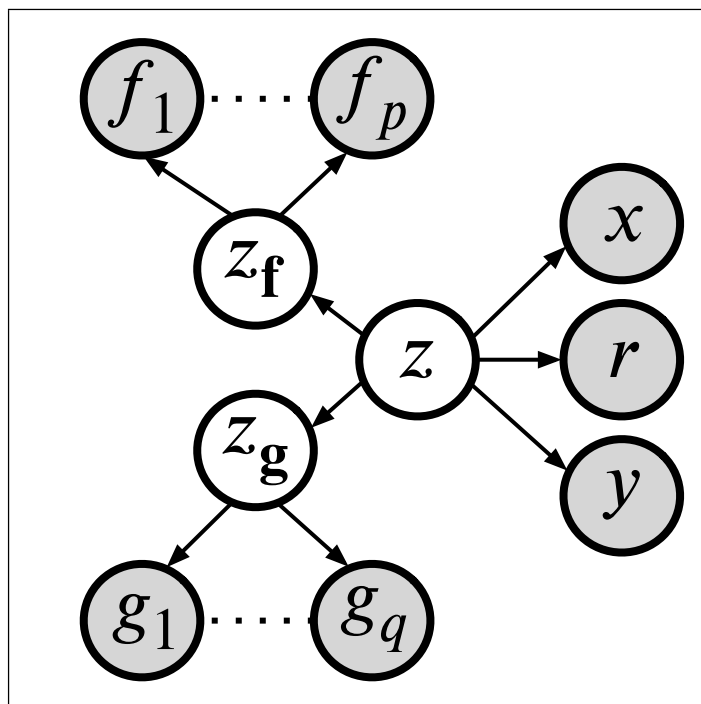


図 13.1: ハイブリッド型のトピックモデル [麻生 06]

10.3 節の共起型確率モデルでは，アイテムの特徴などを容易に導入できる柔軟性があると述べた．こうした拡張には [Popescul 01, Ono 05, 麻生 06, 吉井 06, 川前 07] などがある．一例を挙げよう．図 10.1(a) の利用者 x とアイテム y だけを考慮するモデルに，アイテムの特徴ベクトル $\mathbf{f} = (f_1, \dots, f_p)$ を導入する場合を述べる．これは式 (10.8) に新たな項 $\Pr[\mathbf{f}|z]$ を追加して行う．そして，次の同時確率に対する尤度を最大化することでパラメータは学習できる．

$$\Pr[x, y, \mathbf{f}] = \sum_{z \in \mathcal{Z}} \Pr[z|x] \Pr[y|z] \Pr[\mathbf{f}|z] \Pr[z]$$

特徴ベクトル \mathbf{f} の要素数が一つであれば，離散値なら多項分布で，連続値ならガウス分布などでモデル化できる．もし要素数が二つ以上であれば，多変量ガウス分布などが利用できる．また， z が与えられたときに \mathbf{f} の各要素は独立との単純ベイズの仮定のもとで， $\Pr[\mathbf{f}|z] = \prod_i \Pr[f_i|z]$ などとモデル化してもよい．

文献 [麻生 06] では，アイテムの特徴 \mathbf{f} に加え，利用者の個人属性特徴や，利用意図

などの特徴 $\mathbf{g} = (g_1, \dots, g_q)$ を加えた図 13.1 のようなモデルを提案している。これをまとめると次式のようになる。

$$\Pr[x, y, r, \mathbf{f}, \mathbf{g}] = \Pr[z] \Pr[x|z] \Pr[y|z] \Pr[r|z] \\ \left(\Pr[z_f|z] \prod_i^p \Pr[f_i|z_f] \right) \left(\Pr[z_g|z] \prod_j^q \Pr[g_j|z_g] \right)$$

このように、潜在変数を用いたトピックモデルには、推薦に関係するさまざまな要素を自在に導入できる柔軟性がある。モデルを与えれば、EM アルゴリズムなどによりパラメータを学習できる。いろいろなモデルの中から、集めたテストデータに、機械学習でのモデル選択の手法を適用して、最終的に利用するモデルを最終的に選ばよ。

もう一つの完全結合の手法として、関数モデルにカーネルを導入する [Basilico 04] の方法を紹介する。この方法では、順序回帰問題に帰着させてモデルを獲得する。利用者 x がアイテム y を好む度合いを次の関数でモデル化する。

$$f(x, y) = \sum_{(i,j)} w(i, j) K((x, y), (i, j))$$

ただし、 $w(i, j)$ は重みである。そして、しきい値 θ_k について $f(x, y) \in [\theta_{r-1}, \theta_r)$ を満たすなら、利用者 x のアイテム y に与えた評価値は r と予測する。この重み $w(x, y)$ としきい値 θ_k は、順序回帰問題のためのパーセプトロン型の学習則である PRank アルゴリズム [Crammer 02] で獲得する。そして、 $K((x, y), (i, j))$ はカーネル [麻生 03] であり、協調と内容ベースフィルタリングを結合するとき中心的な役割を果たす。この利用者 x とアイテム y の対についてのカーネルは、適当な仮定のもと、利用者アイテムそれぞれのカーネルの積となる。

$$K((x, y), (i, j)) = K_X(x, i) K_Y(y, j)$$

利用者については、 $x = x'$ なら 1、でなければ 0 をとるカーネル、利用者の個人属性の特徴量 $\mathbf{g}(x)$ に基づくカーネル、利用者 x と i の間の 9.1 節の式 (9.1) の相関をとるカーネルなどを個々に求め、これらのカーネルの和を $K_X(x, x')$ とする。アイテムに

についても同様にカーネルを定義できる。この方法では、カーネルを自由に設計できるので、いろいろな要因を考慮した推薦が可能である。さらに、カーネルトリックが適用できるどのような関数モデルにも適用できるので、多様な応用が可能でもある。

13.2 嗜好予測手法の選択

嗜好予測手法のまとめとして、予測手法の選択指針を、私見をまじえつつ列挙しておく。

最初に、協調フィルタリングか内容ベースフィルタリングかの選択について述べる。6.1節では、両方の手法の長所と短所について論じた。これらのうち、内容ベースにとってはアイテム特徴のデータベースを構築し、維持できるかどうか最大の制約である。この構築・維持コストは、人手を要するため、一般に非常に高い。もう一方の協調フィルタリングでは、嗜好データを収集できるかが大きな制約である。利用者に負担をかけずにその嗜好パターンを巧妙に捉える暗黙的な収集方法か、利用者に評価付けしてもらうよい動機付けが必要になる。これらの両方の制約を満たせるならば、それぞれの長所を生かしたハイブリッドがよいだろう。

2.2節の運用目的別では、概要推薦であれば、非個人化推薦である人気リストや、直接指定型の内容ベースとなり、利用者評価では、システム側は基本的に場を提供するだけで、予測手法は不要である。関連アイテム推薦では、アイテム間型メモリベースの協調フィルタリングや、アイテムの特徴を用いて関連アイテムを見つけるのが一般的である。通知サービスや緊密な個人化では、個人情報蓄積していることが前提なので、協調フィルタリングや間接指定型の内容ベースが利用され、予測手法の役割が重要になる。

ここで、search と experience という推薦対象アイテムの種別 [Senecal 04] についてふれておこう。search アイテムとは、カタログの規格からの洞察によって、どれを購入するかを選択できるもので、実用品に多い。こうしたものには、基本的には直接指定型の内容ベースを利用する。協調フィルタリングや間接指定型の内容ベースは、順

位付けなどのために補助的に利用される。後者の *experience* アイテムは、実際に試さないとは判断できないもので、嗜好品などに多い。こうしたものでは、協調フィルタリングでセレンディピティを重視した推薦が有利となる。

次に、協調フィルタリング手法の中での選択について論じる。まず最初に、どの手法も背後にいろいろな前提がある。例えば、10.4 節の定期購読を対象とした方法は、推薦ごとに利用者が決定をする通常の推薦には適用できない。各手法の前提に、推薦システムを適用しようとする対象が合っているかどうかは定性的に検証すべきである。8.1 節で述べたように、データの更新が頻繁ならメモリベースが有利だが、そうでなければモデルベースが高速な推薦が可能である。

メモリベース法には、利用者間型とアイテム間があるが、予測精度よりも他の規準でどちらかを選択すべきだろう。アイテムの更新が頻繁である場合や、セレンディピティなどについては利用者間型が有利だろう。一方、一時的個人化をしたい場合や、利用者の入れ替わりが頻繁な場合にはアイテム間型が有利だろう。

次にモデルベース法について述べる。クラスタリングモデルは、非常におおまかな推薦しかできない。新規利用者を対象としたおおまかな推薦が主な利用法だろう。他の要因を無視して、予測精度だけを重視するならば、行列分解を用いる関数モデルが、近年の研究では有利であるように著者は思う。確率モデルは、多様な方法が提案されており、いろいろな状況に適応できる。確率モデルの履歴条件型と共起型を比較すると、飽和モデルのパラメータ数はそれぞれ $|\mathcal{R}|^m - 1$ と $nm|\mathcal{R}| - 1$ となる。よって履歴条件型は、評価値が多段階だったりアイテム数が増えたりすると極端に不利になりやすいが、利用者数には依存しないので、利用者は多いほど高精度のモデルが学習できる。一方、共起型は評価値の段階数、アイテム数、および利用者数のいずれに対しても線形で、破綻しにくいといえる。また、暗黙的な評価値で、未評価と否定的評価が区別できないときには共起型が有利である。時系列モデルは、成長に合わせて購入する製品が変わる子供用品など、時間に大きく依存する場合には有効である。ただし、時刻を考慮する分だけモデルは複雑なので、十分なデータは必要になる。嗜好の揺らぎを考慮する手法も幾つか紹介した。嗜好データに揺らぎがあるかどうかは、5.2 節で

述べたように、時間をあけて同じ被験者から嗜好データを集め、整合性が低いかどうかで調べる。

あらゆる状況でうまく予測できるアルゴリズムは存在しないので、サンプルデータを収集して交差確認法で汎化誤差を実験的に求め、手法を比較したり、パラメータを調整する手続きは不可欠である。調整パラメータ数が増えるアルゴリズムを使ったときには、予測精度の向上がわずかならば、パラメータ数の少ない方法を使っておくか、3.1 節で述べたように、データを訓練・確認・テストに 3 分割した厳密な方法で検証することを薦める。

第 IV 部

推薦システムのその他の話題

第 14 章

プライバシー保護協調フィルタリング

協調フィルタリングでは、標本利用者の嗜好データを収集する。このデータが、商品の購入履歴などであった場合は、これらの情報は利用者が秘匿したい個人情報となるであろう。さらに、断片的な情報を集積することで、より重大な個人情報を得ることもできるようになってきている。例えば、Sweeney は、87% の US 在住者が、性別、5 桁郵便番号、生年月日の情報だけで一意に特定できることを示した [Sweeney 00]。映画の評価について、掲示板発言と、これとはまた、独立した推薦システムの評価データのパターンの類似性から、匿名の利用者の ID の対応付けがかなりの割合でできるとの報告もある [Frankowski 06]。現在では、プライバシーは「個人が自らの情報を自分でコントロールできる権利」[沼尾 06] を意味するようになり、ますます重視されるようになってきている。一方、協調フィルタリングはこうした個人情報なしには実現できないので、これらの情報を収集する必要がある。

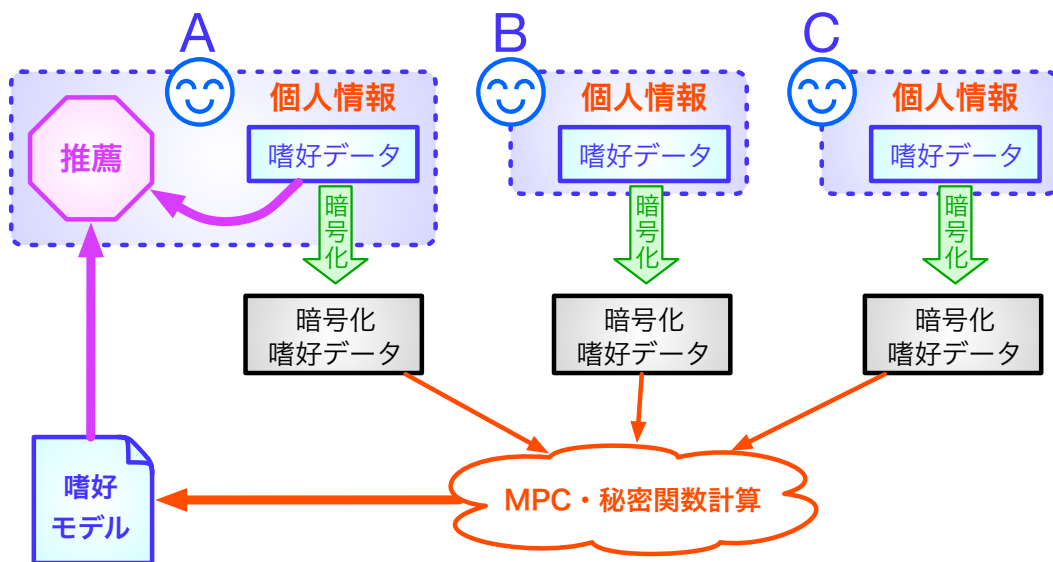
現状では、この個人情報の収集に伴う問題には、プライバシー・ポリシーを公開し、それを遵守することを誓約することで、社会的手段によって対処している。これに対し、技術的手段によってこの問題に対処するのが**プライバシー保護協調フィルタリング** (privacy-preserving collaborative filtering) である。プライバシー保護協調

フィルタリングの技術は、**プライバシー保護データマイニング** (privacy-preserving data mining) [Aggarwal 08, 佐久間 09] と関連が深い。これは、分散環境の各サイトに分割されて保持されているデータがあるとき、それら全てを集めたデータ集合に対するデータマイニング結果を、各サイト内のデータの内容を自分以外のサイトは秘密にしたまま計算する技術である。この計算を実現するアプローチとしては暗号化 [Lindell 00] とランダム化 [Agrawal 00] とを使う方法がある。プライバシー保護協調フィルタリングにおいても、この二つのアプローチがある。これらを順に説明する。

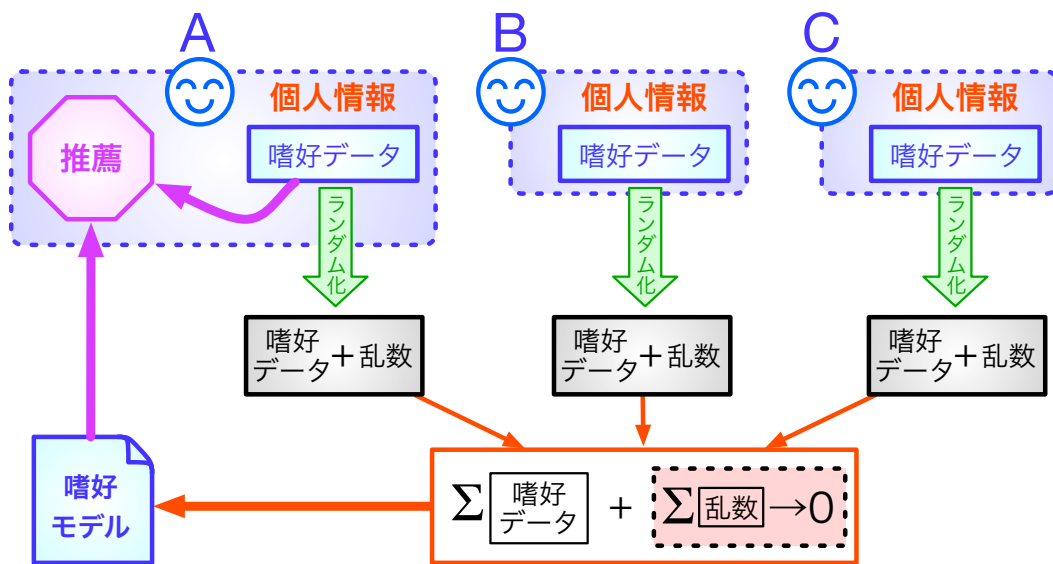
14.1 暗号化による方法

モデルベースの協調フィルタリングでは、標本利用者の嗜好データではなく、それらの規則性を表したモデルさえあれば推薦が可能である。よって、嗜好データそのものを知らせることなくモデルが構築できれば、各利用者がどのような評価をしたかは秘匿できる。また、メモリベースの協調フィルタリングでも、評価値行列を次元削減したものを公開すれば、個々のアイテムへの評価値は秘匿できる。これを、データを暗号化したまま計算するという技術によって実現する。この技術としては、ほとんどの計算が可能だが計算量は多い multiparty secure computation [Yao 86] や、計算の種類ごとに専用の方法が必要だが高速に計算できる**秘密関数計算** (secure function evaluation) [Clifton 02] がある。

このアイデアに基づく枠組みを図 14.1(a) に示す。まず、標本利用者は自身の嗜好データを暗号化して推薦システムに渡す。ここで、渡されたデータは暗号化されているため、個人情報外部には漏れない。これらの暗号化嗜好データに、秘密関数計算などの技術を適用して、暗号化された状態のモデルを獲得する。最後に暗号化モデルを復号して目的の嗜好モデルを得る。ここで、この嗜好モデルは標本利用者全般の嗜好の規則性の情報を表してはいるが、このモデルから個々の標本利用者の嗜好データ自体を復元することはできない。よって、個人の評価情報は漏洩しない。活動利用者は、自身の計算機上で、このモデルと自身の嗜好データから推薦を得ることができる。



(a) 暗号化による方法



(b) ランダム化による方法

図 14.1: プライバシー保護協調フィルタリング

モデルには、利用者間型メモリベースで次元縮約を使うもの [Canny 02a] や、10.2 節の行列分解を使うもの [Canny 02b] などがある。

しかし、現状のプライバシー保護協調フィルタリングでは、社会的手段が補助的に必要となる。計算結果が改竄されていないかを検証するには、半数以上の参加者は、個人情報を明かすほど信頼はできないが、計算の手続きは遵守する程度には信頼できるという semi-honest という前提が必要である。しかし、この前提の技術的手段による保証は難しく、社会的手段によって保証しなければならない。そのため、匿名で参加できる peer-to-peer ネットワークなどでの実現は難しい。ソーシャル・ネットワークなど個人認証がなされるサービスでの推薦や、複数の企業が自身の顧客の情報を秘匿しつつ、共同して推薦をしたい場合などを想定すべきである。

14.2 ランダム化による方法

安全な計算は、暗号化はしているが、個人情報の真の値を外部に出している。そうではなく、真の値ではない値を外部に出す方法が**ランダム化** (randomization) による方法 (図 14.1(b)) である。

原理は単純で、平均値が 0 の正規分布や一様分布に従う乱数を加えてから、自分のデータを外部に出力する。このとき、全参加者は同じ確率分布から独立にサンプリングした乱数を用いることが重要である。ここで、全参加者の出力した値の総和や内積を求める。この値は、真の値の総和に、乱数の総和を加えたものになっている。ここで乱数の総和は、元の分布の平均である 0 に漸近的に近づく。よって、十分な数のデータがあれば、総和や内積が近似的に計算できる。この手法によるプライバシー保護協調フィルタリングは [Polat 03] などで提案されている。この方法は、真の値は外部に出さなくて済む利点があるが、加える乱数の標準偏差が小さいと、真の値が近似的に予測できてしまうとか、厳密な値ではなく、近似値しか計算できないといった欠点がある。

第 15 章

サクラ攻撃

推薦システムの効用について考えてみよう (図 15.1)。利用者は知りたい情報を入手できるようになり、システムに満足するだろう。これにより、システムの運営者には、システムの利用が促進されるという利点がある。では、利用者に推薦される製品や情報の提供者にとってはどうであろうか？ 自身の製品の代わりに競合他社の製品が推薦されて欲しくないとか、たとえ自社の製品が推薦されていても、さらに推薦されるよう望んだりするだろう。これらの要求は必ずしも満たされるとは限らないため、推薦結果を自身に有利にする目的で推薦システムに干渉することが行われ始めている。具

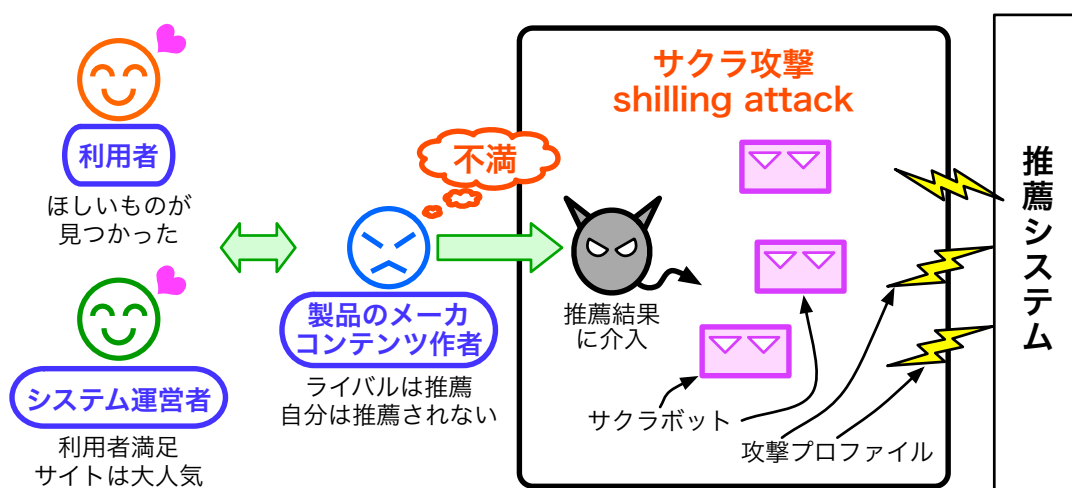


図 15.1: サクラ攻撃を行う動機

体的な事例は [Lam 04] を参照されたい。

こうした行為の一つに**サクラ攻撃** (shilling attack) がある。これは、**サクラボット** (shilling bot) と呼ばれるエージェントプログラムや、雇った人間に、自身に有利な推薦が他の利用者になされるように、推薦システムへ嗜好データを入力させるものである。サクラ攻撃は、利用者にとっては不適切な推薦がなされるため望ましくない。運営者にとってもシステムへの信頼を失わせる行為であるため、サクラ攻撃は排除すべきものである。サクラ攻撃の有効性の検証や、攻撃検出などの対策は、文献 [O'Mahony 04, Lam 04] などによって研究が始められた。サクラ攻撃全般についてのよいまとめとしては文献 [Mobasher 07] がある。以下、サクラ攻撃の構成要素、攻撃戦略の体系的な分類、攻撃への対策を紹介する。

15.1 サクラ攻撃の基本

サクラ攻撃の基本事項をまとめる。攻撃意図とは、攻撃の実行者が推薦をどのように変化させたいかという目的である。

販促攻撃 (push attack) 本来なら推薦されないはずの、自社の製品などのアイテムを推薦されるようにする。

排除攻撃 (nuke attack) 本来なら推薦されるはずの、他社の競合製品などのアイテムを推薦されないようにする。

攪乱攻撃 (random attack) 予測精度を下げさせて、システムへの信頼を失わせる。

このうち前者二つの攻撃目的について研究がなされている。こうした攻撃は、一部の限定された利用者やアイテムに対して集中的に行われる。

一般に、大量の情報、またシステムに密接な情報を利用した攻撃ほど効果的な攻撃が可能になる。特に、利用者の嗜好データに完全にアクセスできると、検出はほぼ不可能である。しかし、この種の情報は、プライバシー保護の観点からも秘匿されて管理されているため、攻撃者が合法的に入手するのは困難である。比較的容易に合法的

に入手できるデータとして、アイテムへの平均評価値がある。これらのデータを公開しているシステムや、利用者そのものでなくても、コメントを付けている編集者や利用者の評価の平均値を出しているものは多い。さらに映画における IMDB [IMDb] のように、アイテムの評価サイトから、代替情報を別途入手できる場合もある。また、推薦アルゴリズムごとに効果的な攻撃の作戦があるので、推薦に用いているアルゴリズムの情報も重要である。一時的な推薦を、協調フィルタリングで行うには、アイテム間型のメモリベース手法が多用されるといったことを用いて、この情報もある程度推定できる。

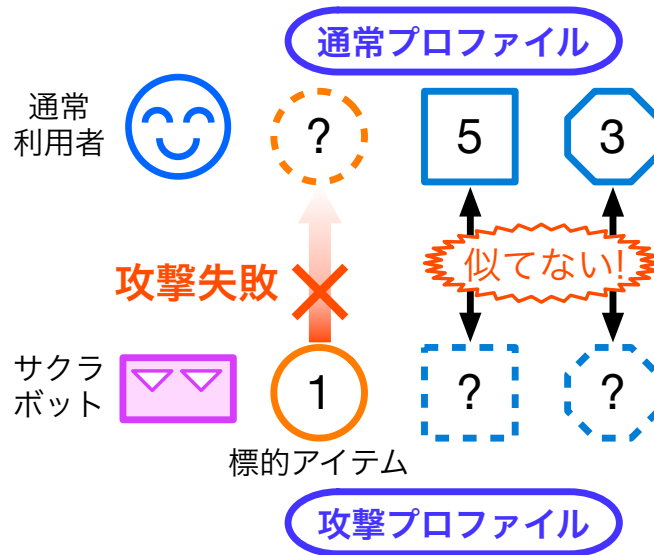
サクラ攻撃の効果は次のような指標によって測る。

予測シフト (Prediction Shift) 攻撃の前後での、予測評価値の変化の全標的アイテム (後述) と全利用者についての平均。予測評価値自体がどれくらい攪乱されたかを測るための指標である。

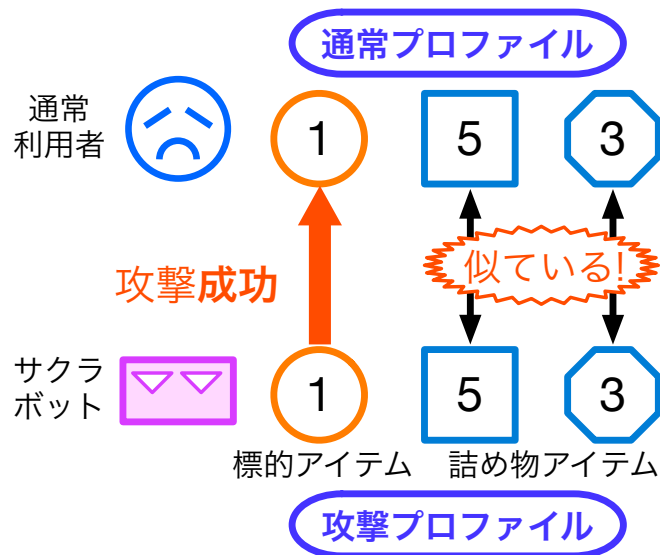
ExpTopN (Expected Top-N Occupancy) 上位 N 個の推薦アイテムに標的アイテムが含まれる個数の、全利用者についての期待値。評価値が変化しても、アイテムの順位に影響するとは限らないので、実際の推薦リストの変化を測るための指標である。

15.2 サクラ攻撃の種類

サクラ攻撃をどのように行うかについて述べる [Lam 04]。図 15.2 は、排除攻撃の様子を示した。まず、図 15.2(a) に注目されたい。上側は、通常利用者の嗜好データ (通常プロファイル) を示した。この通常利用者に推薦されるアイテムを、サクラ攻撃によって変えたいとする。一方、下側は、仮想利用者であるサクラボットが推薦システムに入力する嗜好データで、攻撃プロファイル (attack profile) と呼ぶ。図中には、丸、四角、八角で示した三つのアイテムがあり、各プロファイル中では、これらのアイテムは 5 段階で評価されている。「5」が最高評価、「1」が最低評価、そして「？」が未



(a) 標的アイテムのみ



(b) 標的アイテム+詰め物アイテム

図 15.2: 攻撃プロフィール

評価を示している。この中で、丸で示したアイテムは、排除したい競合製品を表し、これを標的アイテム (target item) と呼ぶ。通常利用者はこの標的アイテムを未評価で、今からこの利用者の標的アイテムへの評価値を予測するとしてよう。

このとき、標的アイテムを推薦されないようにするには図 15.2(a) のように、サクラボットは標的アイテムに最低の評価「1」を与えさえすれば良さそうである。しかし、この攻撃は失敗する。多くのサクラボットが悪い評価を与えることで、このアイテムへの平均的な評価は確かに低下する。だが、協調フィルタリングでは、嗜好パターンが似ている利用者を参考に推薦することを思いだされたい。この利用者の通常プロフィールでは、標的アイテム以外の四角や八角のアイテムも評価されている。一方、攻撃プロフィールでは、丸の標的アイテムのみが評価され、それ以外の四角や八角のアイテムは全て未評価である。そのため、通常プロフィールと攻撃プロフィールは似ていないと判定され、サクラボットとこの利用者の嗜好パターンは違うとみなされる。そのため、標的アイテムの評価は通常利用者の推薦には反映されず、攻撃は失敗する。

そこで、他の利用者の嗜好パターンに攻撃プロフィールを似せるため、図 15.2(b) のように、標的アイテム以外の四角や八角のアイテム群にも評価値を与える。これらのアイテムを詰め物アイテム (filler item) と呼ぶ。これらの、詰め物アイテムへの評価が通常利用者のそれと類似していれば、通常プロフィールと攻撃プロフィールとは類似していると判断される。すると、攻撃プロフィールの標的アイテムへの評価は、通常利用者の推薦に影響し、攻撃は成功する。しかし、通常利用者の嗜好データは入手できない。よって、詰め物アイテムの評価値は、通常利用者の、一般的な評価の傾向に基づいて与える。この与え方の違いによって、次のような攻撃方法がある。

サンプリング攻撃 (sampling attack) 詰め物アイテムに、システムのデータベースから

サンプリングしてきた評価値を与える。統計的手法では検出は不可能と言っても良いが、システム内のデータベースの情報の入手は非常に困難である。

ランダム攻撃 (random attack) ランダムな評価値を、詰め物アイテムに与える

[Lam 04]. 必要な情報はないが、あまり効果はなく発覚し易い。

人気商品攻撃 (bandwagon attack) 良く売れている人気商品の評価は高いことが多く、また多くの利用者が評価している。このことを利用し、人気商品に高い評価値を与えて、攻撃プロファイルの他の利用者への類似性を高める [O'Mahony 04].

平均攻撃 (average attack) 詰め物アイテムに、それらのアイテムへの評価値の平均値を与える [Lam 04]. アイテムへの平均評価値は入手が比較的容易だが、利用者間型のメモリベース手法には効果的である。一方、アイテム間型では、予測シフトで評価すると確かに評価値には影響するが、推薦リスト中でのアイテムの順位を変えるまでにはいならず、ExpTopN への影響は限定的である。

探査攻撃 (probe attack) ダミー利用者を作り、その利用者に推薦されたアイテムの予測評価値を詰め物アイテムへの評価値とする [Mobasher 07]. ダミー利用者と類似した嗜好の利用者にのみ有効な攻撃である。

セグメント攻撃 (segment attack) 映画のジャンルなど、アイテムの分類情報が利用できる場合の販促攻撃に用いる [Burke 05]. 標的アイテムと同じセグメント（分類カテゴリ）のアイテムには、高い評価を与える。同じセグメントのアイテムには、同じような評価がなされやすいという傾向を使う。アイテム間型のメモリベースに対して有効な攻撃である。

愛憎攻撃 (love/hate attack) 排除攻撃専用の手法。標的アイテムには最低の評価を、詰め物アイテムには最高の評価を与える [Burke 06]. 単純な攻撃だがアイテム間型と利用者間型のどちらのメモリベース手法についても効果がある。

15.3 サクラ攻撃に対する防御

作為的な利用者の評価に影響されて、協調フィルタリングが不適切な推薦をしたとしよう。そして、他の利用者がその推薦に従ったとしても、その後、作為のない評価をするので、自律的にこうした攻撃は無力化されるとも考えられていた。しかし、Cosley ら [Cosley 03] はこれに対して否定的な調査結果を報告している。以前に評価したことのあるアイテムについて、以前と同じ、1 段階良い、1 段階悪いの 3 種類

のものを「予測評価」と偽って利用者に提示した。すると、作為的にずらした方向に、アイテムへの利用者の評価が変化した。さらに、未評価のアイテムについて、アルゴリズムができるだけ正確に予測した評価、それより 1 段階良い/悪い評価を利用者に提示すると、やはり、同様の傾向がみられた。このように、心理学における同調 (conformity) [中島 99] と類似した、周りの意見に利用者の意見が『引きずられる』現象が報告されている。そのため、サクラ攻撃に対して推薦システムは自律的に回復することはできず、攻撃を検出して排除する必要がある。

サクラ攻撃は、サンプリング攻撃以外は、真のデータベースの評価値分布と、攻撃プロファイルとの統計的な分布の差をはずれ値検出の技術によって見つけることで検出する。文献 [Zhang 06b] の方法は、最も基本的なもので、モデルから予測される評価値と、実際の評価値の差が標準偏差に対して十分に大きければ攻撃とみなす。ランダム攻撃は比較的容易に検出できるが、平均攻撃では失敗することも多い。また、サクラボットが評価するアイテム数が少ない場合や、標的アイテムが多い場合は検出が困難になると報告している。文献 [Burke 06] では、アイテムに対する平均的な評価からの乖離を、被評価数などで加重した統計量や、ボットの攻撃は多数のアイテムに対して行われることを利用した統計量を提案している。加えて、平均攻撃など特定の攻撃に対する検出方法も検討している。その他、攻撃が特定の時期にまとまって行われることが多いことを利用し、時系列中での検出を試みる研究 [Zhang 06a] もある。これは、あるアイテムについての評価値を、一定間隔の期間ごとに分ける。そして、各期間内の評価値の平均やエントロピーが異常値になることを検定により検出する。文献 [Mehta 08] はサクラ攻撃に対して頑健な、行列分解を用いた推薦アルゴリズムを提案している。この方法では、評価値行列を因子に分解するときに、利用者側の因子はそのまま計算するが、評価パターンが他の利用者と大きく異なっている疑わしい利用者の因子を利用せずにアイテム側の因子を計算する。

第 16 章

推薦システムのその他の問題や 視点

その他の視点に関する研究をまとめて述べる。

16.1 対話型システム

これまでの推薦システムは、過去に入力された嗜好データに基づいて利用者の嗜好を予測し、一度だけ推薦の提示を行う。だが、利用者がその推薦に満足できない場合がある。そうした場合に、活動利用者からフィードバックを得て、その情報を用いてより適切な推薦をする対話的な推薦も研究されている。フィードバックは、アイテムへの評価値を返すアイテムレベルと、アイテムの特徴を指定する特徴レベルとに分けられる。これらを順に紹介する。

16.1.1 アイテムレベルのフィードバック

機械学習のクラス分類の訓練事例では、ラベル付けされている事例の特徴ベクトルは受動的に決められているのが一般的である。そうではなく、能動的に都合のよい特徴ベクトルをもつような事例を指定すると、それに対するラベル情報が得られる状況

を扱うのが能動学習 [渡辺 05] である。この枠組みを導入し、推薦の予測精度が不十分だった場合に、推薦を改善するのに役立つであろうアイテムを利用者に評価してもらうことが考えられる。ではどのアイテムを評価させるのがよいだろうか？ 直観的に、利用者によって評価にばらつきがあり、好みの分かれるアイテムを評価してもらうのはよいだろう。このように、教師情報を得るデータを能動的に選択する機械学習の枠組みを**能動学習** (active learning) という。この枠組みに基づいて、他の利用者の評価値の分散やエントロピーが大きなアイテムを選ぶ方法が提案されている [Kohrs 01]。文献 [Boutillier 03] では、アイテム選択の尺度として、期待情報価値 (expected value of information; EVOI) を提案している。まず、現在の嗜好データで予測したとき、期待的に最も良い評価値は、全未評価アイテム上での、期待評価値の最大値である。ここで、未評価アイテムの一つ x_q に対する評価値を利用者から得ると、この最大期待評価値は変化するはずである。だが、どのような評価値を利用者が返すかは分からないので、評価後の最大期待評価値の、評価値定義域 \mathcal{R} 上の期待値を考える。この期待値から、評価前の最大期待評価値を引いた値を EVOI とする。すると、EVOI は、利用者からアイテム x_q に対する評価値を得たことによる、最大期待評価値が改善する量を表す。よって、この EVOI を最大にする未評価アイテムを活動利用者に評価してもらえばよい。だが、実際には計算量が多いため、この EVOI の値の代わりにその上限を計算して代用する。その他、候補アイテムの評価値が予測どおりであったときと、それとは若干ずれた値であったときの、他の未評価アイテムの予測評価値の変化の大きさが大きいアイテムを選ぶ方法も提案されている [Rubens 07]。この能動的評価の問題点は、利用者が任意のアイテムをすぐに評価でなければ適用できないことである。例えば、音楽であればその場で聞かせて評価させることができるが、映画や書籍の場合では難しい。

純粋な協調フィルタリングでは、コンテキストの情報 (5.5 節) を利用できない。また、コンテキスト自体を特徴として明示することが困難な場合がある。そこで、文献 [Rafter 05] は嗜好データを長期と短期のプロファイルに分けて考える手法を提案している。長期プロファイルとは、現在の推薦のセッションより以前に入力された嗜

好データで、利用者の基礎となる嗜好パターンを表しているとみなす。一方、現在のセッションで提示した推薦リスト中のアイテムに対する嗜好データが短期プロファイルで、現在のコンテキストでの嗜好を示しているとみなす。フィードバックは、目的のアイテムに近いものや、全く当てはまらないものを利用者に指定させる。予測には利用者間型のメモリベース法 (9.1 節) に基づく。長期プロファイルに基づいて活動利用者と標本利用者との類似度を計算する点は同じである。しかし、フィードバックで肯定的だったアイテムを好み、そうでないものを嫌うような標本利用者を重視するように、標本利用者を重み付けする。このようにして、短期プロファイルに暗黙的に示されたコンテキストを反映する推薦が実現される。

文献 [Smyth 03] は、対話的な推薦で、候補アイテムが適切に絞り込まれているかを検証する方法を提案している。アイデアは非常に簡潔で、前回の推薦で肯定的なフィードバックが与えられたアイテムを、次回の推薦リストに混ぜておく。もし、その混ぜておいたそのアイテムが再び選択されるようならば、絞り込みは適切でなかったと判断し、他の方向からの絞り込みに変える。そうでなければ、同じ方向で絞り込みを続ける。

16.1.2 特徴レベルのフィードバック

特徴レベルのフィードバックは、アイテムの特徴を扱うため、内容ベースフィルタリングが対象となる。この種のフィードバックでは、推薦されたアイテムを見て、アイテムの望ましい性質を利用者が示す。例えば、パソコンではメモリが 1GB 以上といった、アイテムの特徴に対する条件の精細化をする。12 章で述べたように、内容ベースの推薦は情報検索と類似しているため、情報検索でフィードバックを扱う手法が、内容ベースの推薦に応用できる。情報検索では、検索質問と関連する語を利用者に提示し、利用者がそれを選択することで、検索質問拡張を実現するものがある。これと同様の手法が、推薦システムでも利用されている [Smyth 04]。

その他、文献 [McSherry 05] では、7.5 節のように、説明を導入してより円滑に対話

的な推薦を行うことを提案している。ある特徴レベルフィードバックを利用者が返すと、次の推薦で候補の数や内容がどのように変わるのかを簡潔に説明し、より適切なフィードバックを得られるようにしている。

16.2 標本利用者の選別

協調フィルタリングでは、他の標本利用者の意見を参照する。このとき、嗜好が似ているだけでなく、該当分野に詳しい利用者の意見を参照する方がよいだろう。こうした試みは、協調フィルタリングが自動化される以前からある [Kautz 97]。

自動化された協調フィルタリングで、標本利用者を目的に応じて選別するアイデアには次のようなものがある。ある標本利用者の嗜好データから、いろいろな活動利用者の評価値を予測する。このとき、より多くの活動利用者の評価値を正確に予測できたとすれば、その標本利用者は他の活動利用者の評価値予測にも有用だろう。文献 [O'Donovan 05] は、このアイデアに基づいた標本利用者の重み付けを提案している。文献 [亀井 04] では、各利用者が個別に文書レポジトリを所有しており、そのレポジトリは公開されている状況を想定する。各文献を特徴ベクトルで表現し、活動利用者の評価とある標本利用者のそれが類似しているとき、その文献に関連した特徴に関して、その標本利用者は信頼できると考える。そして、それと同じ特徴をもつ文献に対する評価では、その標本利用者の意見を重視する。あるカテゴリのアイテムを、より多く評価している利用者は、そのカテゴリのアイテムについて詳しいと仮定してよいだろう。このアイデアに基づき、文献 [Cho 07] では、評価値を予測しようとしている目標アイテムと同じカテゴリのアイテムを多数評価している標本利用者の意見を重視することを提案している。

16.3 推薦システムの運用

その他、推薦システムの運用に関係する問題についてまとめておく。推薦システムを含めたフィルタリング技術の法的側面からの問題点を述べた解説としては [新保 06]

がある。著名な推薦システムの内部アルゴリズムを概説した文献を挙げておく：オンライン小売りサイト Amazon.com[Linden 03]、ニュースの要約サイト Google News[Das 07]、学術文献検索サイト CiteSeer[Lawrence 99]、推薦機能付きセットトップボックス TiVo[Basu 04]。稼働中のシステムの運用に関する情報として、書籍販売サイト bk1 での推薦エンジン AwarenessNet[市川 07] や、携帯向け映画推薦システムのストリートキャッチ方式によるアンケート調査 [小野 07] の結果などがある。

次に、推薦システムに関連の、マーケティング分野の研究について述べる。まず、簡単な市場モデルを示した研究を挙げる。文献 [Fleder 07] では、推薦システムが顧客の購入行動に影響を与え、市場の寡占化が生じるかを 2 個のアイテムがある市場の数理モデルで論じた。購入したことを肯定的、購入しないことを否定的な評価としたとき、購入されたアイテムは推薦されやすくなり、寡占化がすすむ場合が多い。そうならないのは、顧客が推薦を受ける以前は一方のアイテムを強く好んでいるが、推薦にも影響されやすいという状況だけに限られる。文献 [Bergemann 06] では、1 回目的一方の店で購入すると推薦を受けられるが、他方の店で購入すると受けられないモデルの均衡状態を求めている。このモデルで、推薦を受けられることの付加的な価値が、二つの店舗のシェアや顧客が購入するアイテムに与える影響を論じている。推薦の正確さと、好みの方の商品を選択することが顧客にとってどれくらい重要であるかによって、4 種類の異なる均衡状態があることを示している。

次に、推薦が顧客の意志決定に与える影響をアンケート調査した研究を示す。文献 [Senecal 04] では、仕様だけで好みの判断ができる search 型の商品群と、試さない判断できない experience 型の違いについて調査し、experience 型の商品群の方がより推薦に従うと報告している。他の顧客の推薦より、推薦システムは信頼されていないにもかかわらず、実際の影響は推薦システムの方が強かったとも報告している。また、販売業者から独立した組織の推薦であるかどうかは、信頼性や意志決定に影響しないことを報告している。顧客はまだ知らないアイテムを推薦するとき、同時に提示する情報の影響について [Cooke 02] は報告している。顧客が好きなアイテムを同じ画面で共に示すと、知らないアイテムも同様にみなされて、好まれる。だが、この効

果は、別画面に表示するだけで失われてしまう。知らないアイテムに固有の情報与える（サンプル曲を聴かせるなど）と、差異が意識され、知らない推薦への魅力が下がるとも報告している。なお、これらは、実際のシステムの推薦ではなく、推薦システムからの出力であることにして被験者に示した場合の調査結果である。

推薦システムによって、利用者の嗜好を逆転さえできることは、上記のような研究に加え [Cosley 03] でも指摘されている。だが、こうした行為を長期的に継続したときの影響は調査されていない。著者は、推薦システムは [Ben Schafer 01] にもあるように販売側ではなく、顧客側に立つことは必要な条件だと考えている。なぜなら、こうした利用者の嗜好のねじ曲げは、長期的には、推薦システムへの信頼を損なう危険が多いだろう。こうしたことをせず、利用者の意向に沿うことに徹しても [Ben Schafer 01] にあるような三つの効用があるだろう。第 1 に、顧客が真に好むものを積極的に提示することで、閲覧者を購入者に変える。第 2 に関連商品を推薦して cross-sell を増加させる。最後に、顧客側の推薦を続けることによって長期的な顧客ロイヤリティを構築する。推薦システムは、どんなに売れない商品を売ってしまう魔法の杖ではなく、逆に売れ筋商品の寡占を加速する場合もあることは意識しておくべきだろう。

16.3.1 推薦の品質

予測精度以外の規準も考慮した推薦の品質 [McNee 06a] に関する研究を紹介する。

予測精度以外の評価指標の多様な規準の必要性は [McNee 06b] などで指摘されている。こうした基準には、アイテムの特徴 [Ziegler 05] や、単純な予測器 [村上 07] を用いた意外性の規準がある。また、全利用者について推薦されるアイテムの種類分布の Gini 係数を用いて推薦の多様性の評価する [Fleder 07] も、アイテム集合全体を評価する点で興味深い。現在は推薦するアイテムを個々に判断しているが、推薦リスト全体で最適化することを今後は重視してゆくべきだろう [McNee 06a]。

データ入力段階では、利用者の要求をうまく獲得することが重要だろう。まず、好きかどうかだけではなく、複数の評価項目について質問することで、どういう点で良

かったのかを知る方法がある。例えば、映画について、全体評価の他、脚本や配役などの評価も利用する [Adomavicius 07] や、好みに加えて知っているかどうかも尋ねる [清水 07] などの研究がある。暗黙的な獲得でも、閲覧、サンプル視聴、購入の行動を区別する [Cho 07] などの研究もある。効果的な評価項目について検討する余地は広い。

嗜好の予測段階について述べる。個人ではなく複数の利用者から構成されるグループに対する**グループ推薦** (group recommendation) がある [O'Connor 01, 山根 06, Seko 11, Baltrunas 10, Berkovsky 10]。全利用者に対して平均的によい推薦や、だれにもあまり嫌われないような推薦など、いろいろな選択肢が考えられる。

第 V 部

まとめ

第 17 章

参考資料の紹介

推薦システムについてさらに知るには以下のような資料がある。

17.1 代表的な文献, 解説記事, および書籍

17.1.1 英語

Using Collaborative Filtering to Weave an Information Tapestry (1992)

手動での検索ではあるが, 初めて “collaborative filtering” の概念を提案した [Goldberg 92]

GroupLens: An Open Architecture for Collaborative Filtering of Netnews (1994) と Social Information Filtering: Algorithms for Automating “Word of Mouth” (1995)

現在の推薦システムに繋がる, 最初の自動化した協調フィルタリングによる推薦システムである GroupLens[Resnick 94] と Ringo[Shardanand 95]

Recommender Systems (1997)

『推薦システム』という用語を定着させた Communications of ACM の特集記事 [Resnick 97]

Empirical Analysis of Predictive Algorithms for Collaborative Filtering (1998)

初期の推薦アルゴリズムのサーベイと予測精度の比較 [Breese 98]

E-Commerce Recommendation Applications (2001)

電子商取引サイトでの推薦システムの運用について述べた初期の文献では

あるが、ここで指摘された多くの考えは現在においても示唆に富んでいる
[Ben Schafer 01]

Evaluating Collaborative Filtering Recommender Systems (2004)

推薦システムの評価について「良い推薦とは」ということに深い考察がなされている [Herlocker 04]

Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions (2005)

推薦システムの基本的なアルゴリズムを俯瞰し、拡張的な推薦タスクに言及
[Adomavicius 05]

Attacks and Remedies in Collaborative Recommendation (2007)

サクラ攻撃全般について概要を知るのによい [Mobasher 07]

A Survey of Accuracy Evaluation Metrics of Recommendation Tasks (2009)

推薦のタスクごとに適切な評価尺度を選ぶべきことを実験的に示した
[Gunawardana 09]

17.1.2 日本語

情報推薦・情報フィルタリングのためのユーザプロファイリング技術 (2004)

嗜好データの収集全般についてまとめている [土方 04]

特集『情報のフィルタリング』(2006)

有害情報のフィルタリングと表現の自由など法的観点から、推薦システムの概要まで情報フィルタリングについて幅広く扱う [川瀬 06]

特集『利用者の好みをとらえ活かす – 嗜好抽出技術の最前線 –』

推薦システム全般を簡潔にまとめた [土方 07b] 他いくつかの解説記事を含む
[土方 07a]

推薦システムのアルゴリズム (2007–2008)

本稿はこの拙著に訂正・加筆を行ったものである [神嶌 07, 神嶌 08a, 神嶌 08b]

情報推薦システム入門 — 理論と実践

ややヒューマン・コンピュータ・インターフェースよりの解説 [Jannach 12]

17.2 チュートリアル講演

国際会議などで行われたチュートリアルを紹介する。

- Grouplens 法とインターフェース関係 [Konstan 03]
- 予測精度以外に考慮すべき多様性などの要素 [McNee 06a, McNee 06b]
- 推薦システムでユーザインターフェースが果たす役割 [Konstan 10]
- 推薦システムの評価 [Shani 10]
- 推薦システムと Web コンテンツ最適化の高度なアルゴリズム [Agarwal 10]
- フィルターバブル問題に対するパネル [Resnick 11]
- 推薦と情報検索の多様性 [Hurley 11]
- ソーシャル推薦 [Jamali 11]
- 利用者インタフェースの評価 [Knijnenburg 12]
- 推薦システムの AB テストなど実地検証 [Kohavi 12]
- 推薦システム全般 [Jannach 13]

17.3 Web サイト

推薦システムに関連する Web サイトを紹介する

ACM Recommender Systems 2007 年から始まった推薦システムの国際会議
[RecSys]

ACM RecSysWiki 国際会議 Recommender Systems のコミュニティにより運営され
ている Wiki[RecSysWiki]

GroupLens Research Lab. MovieLens の実験システム, プロジェクトの参考文献, 代
表的なテストデータなどを参照できる [Grouplens]

Netflix Prize オンラインビデオ配信の Netflix 社が 100 万ドルの賞金をかけて 2006～

2011 年の間に行ったコンペティションで、推薦の予測精度の研究に貢献した
[NetflixPrize 06]

17.4 テスト用データ

アルゴリズムを検証するために利用できるベンチマークデータを紹介する。なお、推薦システム研究の初期によく利用された、映画評価データ EachMovie や Netflix のコンペティションのデータなどもあるが、現在は配布が停止されているので、これを利用するのは避けた方がよいだろう。

Movielens 映画推薦システム MovieLens (最も良く使われているベンチマーク)

<http://grouplens.org/datasets/movielens/>

SUSHI Preference Data Sets 著者が収集した寿司に対する嗜好データ、順位法と採点法の両方の評価

<http://www.kamishima.net/sushi/>

Jester アメリカンジョークの推薦、評価スケールが 100 段階なのが特徴的

<http://www.ieor.berkeley.edu/~goldberg/jester-data/>

BookCrossing 書籍コミュニティサイト

<http://www2.informatik.uni-freiburg.de/~chiegler/BX/>

Cuban cigars 葉巻販売サイトの商品の閲覧と購入、サイトの閲覧行動

<http://isl.ifit.uni-klu.ac.at/>

Audioscrobbler 現在は Last.fm の一部となった Audioscrobbler の音楽データ

http://www-etud.iro.umontreal.ca/~bergstrj/audioscrobbler_data.html

Flixster M. Jammali が収集した、アイテムの評価に加えて、利用者間の信頼関係データを含むソーシャル推薦用

<http://www.sfu.ca/~sja25/datasets/>

BibSonomy 論文に関するソーシャルブックマークサイト

<http://www.kde.cs.uni-kassel.de/bibsonomy/dumps>

MovieTweatings Twitter のツイートから映画の評価データを収集したもの

<https://github.com/sidooms/MovieTweatings>

WikiLens タグ・評価付きの Wiki のデータ

<http://grouplens.org/datasets/wikilens/>

HetRec 2011 RecSys2011 の併設ワークショップでのデータ, 評価の他に追加的な情報がありコンテキスト考慮型推薦などの実験に

<http://grouplens.org/datasets/hetrec-2011/>

Yelp 商品の評価サイト

http://www.yelp.com/dataset_challenge/

17.5 ソフトウェア

主なオープンソースの推薦システム

LensKit Grouplens 研究グループによる java で実装されたライブラリ

<http://lenskit.grouplens.org/>

MyMediaLite Microsoft 社の .NET 環境下で動作する

<http://mymedialite.net/>

GraphLab 線形代数のライブラリだが, 行列分解型の推薦アルゴリズムを多く実装している

<http://graphlab.org>

Mahout MapReduce 上で動作する大規模用

<http://mahout.apache.org/>

mrec 文献管理サイト Mendeley が公開している Python で実装されたライブラリ

<https://github.com/mendeley/mrec>

参考文献

- [Abe 99] N. Abe and A. Nakamura. Learning to optimally schedule internet banner advertisements. In *Proc. of the 16th Int'l Conf. on Machine Learning*, pp. 12–21, 1999.
- [Adomavicius 05] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowledge and Data Engineering*, Vol. 17, No. 6, pp. 734–749, 2005.
- [Adomavicius 07] G. Adomavicius and Y. Kwon. New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems*, Vol. 22, No. 3, pp. 48–55, 2007.
- [Agarwal 09] D. Agarwal, B.-C. Chen, and P. Elango. Explore/exploit schemes for web content optimization. In *Proc. of the 9th IEEE Int'l Conf. on Data Mining*, pp. 1–10, 2009.
- [Agarwal 10] D. Agarwal and B.-C. Chen. Recommender problems for web applications. The 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, Tutorial, 2010.
- [Aggarwal 99] C. C. Aggarwal, J. L. Wolf, K.-L. Wu, and P. S. Yu. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In *Proc. of the 5th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 201–212, 1999.
- [Aggarwal 08] C. C. Aggarwal and P. S. Yu. *Privacy Preserving Data Mining: Models and Algorithms*. Advances in Database Systems. Springer-Verlag, 2008.
- [Agrawal 00] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proc of The ACM SIGMOD Int'l Conf. on Management of Data*, pp. 439–450, 2000.
- [Agresti 03] A. Agresti. カテゴリカルデータ解析入門. サイエンス社, 2003. (渡邊 裕之他訳).
- [Ali 04] K. Ali and W. van Stam. Tivo: Making show recommendations using a distributed collaborative filtering architecture. In *Proc. of the 10th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 394–401, 2004.
- [Anderson 04] C. Anderson. The long tail. *Wired*, 2004. (<http://www.wired.com/wired/archive/12.10/tail.html>).
- [Arrow 70] K. J. Arrow. *Social Choice and Individual Values*. Yale University Press, second edition, 1970.
- [麻生 03] 麻生英樹, 津田宏治, 村田昇. パターン認識と学習の統計学 — 新しい概念と手法, 統計科学のフロンティア, 第6巻. 岩波書店, 2003.
- [麻生 06] 麻生英樹, 小野智弘, 本村陽一, 黒川茂莉, 櫻井彰人. 協調フィルタリングと属性ベース

- フィルタリングの統合について. 電子情報通信学会技術研究報告, NC 2006-54, 2006.
- [Auer 02] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, Vol. 47, pp. 235–256, 2002.
- [Balabanović 97] M. Balabanović and Y. Shoham. Fab: Content-based, collaborative recommendation. *Communications of the ACM*, Vol. 40, No. 3, pp. 66–72, 1997.
- [Baltrunas 10] L. Baltrunas, T. Makcinskas, and F. Ricci. Group recommendations with rank aggregation and collaborative filtering. In *Proc. of the 4th ACM conference on Recommender systems*, pp. 119–126, 2010.
- [Basilico 04] J. Basilico and T. Hofmann. Unifying collaborative and content-based filtering. In *Proc. of the 21st Int'l Conf. on Machine Learning*, pp. 65–72, 2004.
- [Basu 01] C. Basu, H. Hirsh, W. W. Cohen, and C. Nevill-Manning. Technical paper recommendation: A study in combining multiple information sources. *Journal of Artificial Intelligence Research*, Vol. 14, pp. 231–252, 2001.
- [Basu 04] S. Basu, M. Bilenko, and R. J. Mooney. A probabilistic framework for semi-supervised clustering. In *Proc. of the 10th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 59–68, 2004.
- [Bell 07a] R. M. Bell and Y. Koren. Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *Proc. of the 7th IEEE Int'l Conf. on Data Mining*, pp. 43–52, 2007.
- [Bell 07b] R. M. Bell, Y. Koren, and C. Volinsky. Modeling relationships at multiple scales to improve accuracy of large recommender systems. In *Proc. of the 13th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 95–104, 2007.
- [Ben Schafer 01] J. Ben Schafer, J. A. Konstan, and J. Riedl. E-commerce recommendation applications. *Data Mining and Knowledge Discovery*, Vol. 5, pp. 115–153, 2001.
- [Bergemann 06] D. Bergemann and D. Ozmen. Optimal pricing with recommender system. In *ACM Conference on Electronic Commerce*, pp. 43–51, 2006.
- [Berkovsky 10] S. Berkovsky and J. Freyne. Group-based recipe recommendations: Analysis of data aggregation strategies. In *Proc. of the 4th ACM conference on Recommender systems*, pp. 111–118, 2010.
- [Bishop 06] C. M. Bishop. *Pattern Recognition And Machine Learning*. Information Science and Statistics. Springer, 2006.
- [Bishop 08] C. M. Bishop. パターン認識と機械学習 上下 — ベイズ理論による統計的予測. シュプリンガー・ジャパン, 2007–2008. [監訳: 元田 浩 他; 翻訳: 神嶋 敏弘 他].
- [Blei 03] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, Vol. 3, pp. 993–1022, 2003.
- [Borda 95] J.-C. de Borda. On elections by ballot (1784). In I. McLean and A. B. Urken, editors, *Classics of Social Choice*, chapter 5, pp. 81–89. The University of Michigan Press, 1995.
- [Bottou 16] L. Bottou, F. E. Curtis, and J. Nocedal. Stochastic gradient methods for large-scale machine learning. The 33rd Int'l Conf. on Machine Learning, Tutorial, 2016.

- [Boutillier 03] C. Boutilier, R. S. Zemel, and B. Marlin. Active collaborative filtering. In *Uncertainty in Artificial Intelligence 19*, pp. 98–106, 2003.
- [Breese 98] J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Uncertainty in Artificial Intelligence 14*, pp. 43–52, 1998.
- [Brutlag 09] J. Brutlag. Speed matters for Google web search. Technical report, Google, Inc., 2009. (<http://code.google.com/speed/files/delayexp.pdf>).
- [Brynjolfsson 06] E. Brynjolfsson, Y. J. Hu, and M. D. Smith. From niches to riches: Anatomy of the long tail. *MIT Sloan Management Review*, Vol. 47, No. 4, pp. 67–71, 2006.
- [Burke 02] R. Burke. Hybrid recommender systems: Survey and experiments. *User-Modeling and User-Adapted Interactions*, Vol. 12, No. 4, pp. 331–370, 2002.
- [Burke 05] R. Burke, B. Mobasher, R. Bhaumik, and C. Williams. Segment-based injection attacks against collaborative filtering recommender systems. In *Proc. of the 5th IEEE Int'l Conf. on Data Mining*, pp. 577–580, 2005.
- [Burke 06] R. Burke, B. Mobasher, C. Williams, and R. Bhaumik. Classification features for attack detection in collaborative recommender systems. In *Proc. of the 12th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 542–547, 2006.
- [Canny 02a] J. Canny. Collaborative filtering with privacy. In *Proc. of the 2002 IEEE Symposium on Security and Privacy*, pp. 45–57, 2002.
- [Canny 02b] J. Canny. Collaborative filtering with privacy via factor analysis. In *Proc. of the 25th Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 238–245, 2002.
- [Celma 08] Ò. Celma and P. Cano. From hits to niches?: or how popular artists can bias music recommendation and discovery. In *Proc. of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*, 2008.
- [Cho 07] J. Cho, K. Kwon, and Y. Park. Collaborative filtering using dual information sources. *IEEE Intelligent Systems*, Vol. 22, No. 3, pp. 30–38, 2007.
- [Claypool 99] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin. Combining content-based and collaborative filters in an online newspaper. In *ACM SIGIR Workshop on Recommender Systems: Algorithms and Evaluation*, 1999.
- [Clifton 02] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu. Tools for privacy preserving distributed data mining. *SIGKDD Explorations*, Vol. 4, No. 2, pp. 28–34, 2002.
- [Condliff 99] M. K. Condliff, D. M. Lambda, D. D. Lewis, and C. Posse. Bayesian mixed-effects models for recommender systems. In *ACM SIGIR Workshop on Recommender Systems: Algorithms and Evaluation*, 1999.
- [Cooke 02] A. D. J. Cooke, H. Sujan, M. Sujan, and B. A. Weitz. Marketing the unfamiliar: The role of context and item-specific information in electronic agent recommendations. *Journal of Marketing Research*, Vol. 39, No. 4, pp. 488–497, 2002.
- [Cosley 03] D. Cosley, S. K. Lam, I. Albert, J. A. Konstan, and J. Riedl. Is seeing believing?

- how recommender interfaces affect users' opinions. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, pp. 585–592, 2003.
- [Crammer 02] K. Crammer and Y. Singer. Pranking with ranking. In *Advances in Neural Information Processing Systems 14*, pp. 641–647, 2002.
- [Das 07] A. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: Scalable online collaborative filtering. In *Proc. of the 16th Int'l Conf. on World Wide Web*, pp. 271–280, 2007.
- [Dempster 77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of The Royal Statistical Society (B)*, Vol. 39, No. 1, pp. 1–38, 1977.
- [Dwork 01] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proc. of the 10th Int'l Conf. on World Wide Web*, pp. 613–622, 2001.
- [Dwork 06] C. Dwork. Differential privacy. In *Proc. of the 33rd Int'l Colloquium on Automata, Languages, and Programming*, pp. 1–12, 2006. [LNCS 4052].
- [Fleder 07] D. Fleder and K. Hosanagar. Recommender systems and their impact on sales diversity. In *ACM Conference on Electronic Commerce*, pp. 192–199, 2007.
- [Frankowski 06] D. Frankowski, D. Cosley, S. Sen, L. Terveen, and J. Riedl. You are what you say: Privacy risks of public mentions. In *Proc. of the 29th Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 565–572, 2006.
- [Freund 96] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Proc. of the 13th Int'l Conf. on Machine Learning*, pp. 148–156, 1996.
- [Freund 98] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. In *Proc. of the 15th Int'l Conf. on Machine Learning*, pp. 170–178, 1998.
- [フロインド 99] フロインド Y., シャピリ R., 阿部直樹. ブースティング入門. 人工知能学会誌, Vol. 14, No. 5, pp. 771–780, 1999.
- [Freund 03] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, Vol. 4, pp. 933–969, 2003.
- [George 05] T. George and S. Merugu. A scalable collaborative filtering framework based on co-clustering. In *Proc. of the 5th IEEE Int'l Conf. on Data Mining*, pp. 625–628, 2005.
- [Goldberg 92] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, Vol. 35, No. 12, pp. 61–70, 1992.
- [Good 99] N. Good, J. Ben Schafer, J. A. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl. Combining collaborative filtering with personal agents for better recommendations. In *Proc. of the 16th National Conf. on Artificial Intelligence*, pp. 439–446, 1999.
- [Grouplens] GroupLens research lab, university of Minnesota. (<http://www.grouplens.org/>).
- [Gunawardana 09] A. Gunawardana and G. Shani. A survey of accuracy evaluation metrics of recommendation tasks. *Journal of Machine Learning Research*, Vol. 10, pp. 2935–2962,

- 2009.
- [Han 11] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers, third edition, 2011.
- [Hastie 09] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer-Verlag, second edition, 2009.
- [Häubl 01] G. Häubl and K. B. Murray. Recommending or persuading? the impact of a shopping agent's algorithm on user behavior. In *ACM Conference on Electronic Commerce*, pp. 163–170, 2001.
- [Heckerman 00] D. Heckerman, D. M. Chickering, C. Meek, R. Rounthwaite, and C. Kardia. Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research*, Vol. 1, pp. 49–75, 2000.
- [Herlocker 99] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *Proc. of the 22nd Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 230–237, 1999.
- [Herlocker 00] J. L. Herlocker, J. A. Konstan, and J. Riedl. Explaining collaborative filtering recommendations. In *Proc. of the Conf. on Computer Supported Cooperative Work*, pp. 241–250, 2000.
- [Herlocker 04] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. on Information Systems*, Vol. 22, No. 1, pp. 5–53, 2004.
- [土方 04] 土方嘉徳. 情報推薦・情報フィルタリングのためのユーザプロファイリング技術. 人工知能学会誌, Vol. 19, No. 3, pp. 365–372, 2004.
- [土方 07a] 土方嘉徳ほか. 特集『利用者の好みをとらえ活かす – 嗜好抽出技術の最前線 –』. 情報処理, Vol. 48, No. 9, pp. 955–1007, 2007.
- [土方 07b] 土方嘉徳. 嗜好抽出と情報推薦技術. 情報処理, Vol. 48, No. 9, pp. 957–965, 2007.
- [Hill 95] W. Hill, L. Stead, M. Rosenstein, and G. Furnas. Recommending and evaluating choices in a virtual community of use. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, pp. 194–201, 1995.
- [Hofmann 99a] T. Hofmann. Probabilistic latent semantic analysis. In *Uncertainty in Artificial Intelligence 15*, pp. 289–296, 1999.
- [Hofmann 99b] T. Hofmann. Probabilistic latent semantic indexing. In *Proc. of the 22nd Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 50–57, 1999.
- [Hofmann 99c] T. Hofmann and J. Puzicha. Latent class models for collaborative filtering. In *Proc. of the 16th Int'l Joint Conf. on Artificial Intelligence*, pp. 688–693, 1999.
- [Hofmann 03] T. Hofmann. Collaborative filtering via gaussian probabilistic latent semantic analysis. In *Proc. of the 26th Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 259–266, 2003.

- [Hurley 11] N. Hurley. Towards diverse recommendation. In *RecSys Workshop: Novelty and Diversity in Recommender Systems, Keynote*, p. 1, 2011.
- [市川 07] 市川祐介. 協調フィルタリングを用いたレコメンドサービスの導入事例と課題. *情報処理*, Vol. 48, No. 9, pp. 972–978, 2007.
- [IMDb] IMDb: The internet movie database. (<http://imdb.com/>).
- [石井 98] 石井健一郎, 前田英作, 上田修功, 村瀬洋. わかりやすいパターン認識. オーム社, 1998.
- [Iwata 06] T. Iwata, K. Saito, and T. Yamada. Recommendation method for extending subscription periods. In *Proc. of the 12th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 574–579, 2006.
- [岩田 07] 岩田具治, 山田武士, 上田修功. 購買順序を考慮した協調フィルタリング. *電子情報通信学会技術研究報告*, AI 2007–3, 2007.
- [Jamali 11] M. Jamali and M. Ester. Mining social networks for recommendation. The 11th IEEE Int'l Conf. on Data Mining, Tutorial, 2011.
- [Jannach 12] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich. 情報推薦システム入門 — 理論と実践. 共立出版, 2012. [監訳] 田中 克己, 角谷 和俊.
- [Jannach 13] D. Jannach and G. Friedrich. Recommender systems. The 23rd Int'l Joint Conf. on Artificial Intelligence, Tutorial, 2013.
- [Jin 03] R. Jin, L. Si, and C.-X. Zhai. Preference-based graphic models for collaborative filtering. In *Uncertainty in Artificial Intelligence 19*, pp. 329–336, 2003.
- [Joachims 02] T. Joachims. Optimizing search engines using clickthrough data. In *Proc. of the 8th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 133–142, 2002.
- [亀井 04] 亀井剛次, 船越要, 赤埴淳一, 佐藤哲司. 個人の推薦に基づく個人間情報共有モデル. *人工知能学会論文誌*, Vol. 19, No. 5, pp. 540–547, 2004.
- [Kamishima] T. Kamishima. Sushi preference data sets. (<http://www.kamishima.net/sushi/>).
- [神嶋 03a] 神嶋敏弘. データマイニング分野のクラスタリング手法 (1) — クラスタリングを使ってみよう! —. *人工知能学会誌*, Vol. 18, No. 1, pp. 59–65, 2003.
- [Kamishima 03b] T. Kamishima. Nantonac collaborative filtering: Recommendation based on order responses. In *Proc. of The 9th Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 583–588, 2003.
- [神嶋 04] 神嶋敏弘. なんとなく協調フィルタリング — 順序応答に基づく推薦. *人工知能学会研究会資料*, SIG-KBS–A304-37, 2004.
- [Kamishima 06] T. Kamishima and S. Akaho. Nantonac collaborative filtering — recommendation based on multiple order responses. In *Proc. of the 1st Int'l Workshop on Data Mining and Statistical Science*, pp. 117–124, 2006.
- [神嶋 07] 神嶋敏弘. 推薦システムのアルゴリズム (1). *人工知能学会誌*, Vol. 22, No. 6, pp. 826–837, 2007.
- [神嶋 08a] 神嶋敏弘. 推薦システムのアルゴリズム (2). *人工知能学会誌*, Vol. 23, No. 1, pp. 89–103, 2008.

- [神島 08b] 神島敏弘. 推薦システムのアルゴリズム (3). 人工知能学会誌, Vol. 23, No. 2, pp. 248–263, 2008.
- [Kautz 97] H. Kautz, B. Selman, and M. Shah. Combining social networks and collaborative filtering. *Communications of the ACM*, Vol. 40, No. 3, pp. 63–65, 1997.
- [川前 07] 川前徳章, 高橋克巳, 山田武士. ユーザの興味とオブジェクトのトピックに着目した情報検索モデル. 電子情報通信学会論文誌 D, Vol. J90-D, No. 10, pp. 2746–2754, 2007.
- [川瀬 06] 川瀬直人, 大田原章雄, 木下和彦, 野本東, 高島有次. 特集『情報のフィルタリング』. 情報の科学と技術, Vol. 56, No. 10, pp. 445–481, 2006.
- [北 99] 北研二. 確率的言語モデル, 言語と計算, 第 4 巻. 東京大学出版会, 1999.
- [Knijnenburg 12] B. Knijnenburg. Conducting user experiments in recommender systems. In *The 6th ACM Conf. on Recommender Systems, Tutorial*, 2012. (<http://www.slideshare.net/usabart/tutorial-on-conducting-user-experiments-in-recommender-systems>).
- [Kohavi 12] R. Kohavi. Online controlled experiments: Introduction, learnings, and humbling statistics. The 6th ACM Conf. on Recommender Systems, Keynote, 2012.
- [Kohrs 01] A. Kohrs and B. Merialdo. Improving collaborative filtering for new-users by smart object selection. In *proc. of Int'l Conf. on Media Features*, 2001.
- [Konstan 03] J. A. Konstan and J. Riedl. Recommender systems: Collaborating in commerce and communities. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems, Tutorial*, 2003.
- [Konstan 10] J. A. Konstan. HCI for recommender systems: An introduction. The 4th ACM Conf. on Recommender Systems, Tutorial, 2010.
- [Koren 08] Y. Koren. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proc. of the 14th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 426–434, 2008.
- [Koren 09] Y. Koren. Collaborative filtering with temporal dynamics. In *Proc. of the 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 447–455, 2009.
- [Lam 04] S. (T.) K. Lam and J. Riedl. Shilling recommender systems for fun and profit. In *Proc. of the 13th Int'l Conf. on World Wide Web*, pp. 393–402, 2004.
- [Lawrence 99] S. Lawrence, C. Lee Giles, and K. Bollacker. Digital libraries and autonomous citation indexing. *IEEE Computer*, Vol. 32, No. 6, pp. 67–71, 1999.
- [Lawrence 01] R. D. Lawrence, G. S. Almasi, V. Kotlyar, M. S. Viveros, and S. S. Duri. Personalization of supermarket product recommendations. *Data Mining and Knowledge Discovery*, pp. 11–32, 2001.
- [Lee 01] W. S. Lee. Collaborative learning for recommender systems. In *Proc. of the 18th Int'l Conf. on Machine Learning*, pp. 314–321, 2001.
- [Levy 10] M. Levy and K. Bosteels. Music recommendation and the long tail. In *WOMRAD 2010: RecSys 2010 Workshop on Music Recommendation and Discovery*, 2010.
- [Li 10] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proc. of the 19th Int'l Conf. on World Wide*

- Web*, pp. 661–670, 2010.
- [Lindell 00] Y. Lindell and B. Pinkas. Privacy preserving data mining. In *Advances in Cryptology – CRYPTO2000*, pp. 36–54, 2000. [LNCS 1880].
- [Linden 03] G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, Vol. 7, No. 1, 2003.
- [Luaces 04] O. Luaces, G. F. Bayón, J. R. Quevedo, J. Díez, J. J. del Coz, and A. Bahamonde. Analyzing sensory data using non-linear preference learning with feature subset selection. In *Proc. of the 15th European Conf. on Machine Learning*, pp. 286–297, 2004. [LNCS 3201].
- [Ma 08] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization. In *Proc. of the 17th ACM Conf. on Information and Knowledge Management*, pp. 931–940, 2008.
- [Maes 94] P. Maes. Agents that reduce work and information overload. *Communications of ACM*, Vol. 37, No. 7, pp. 30–40, 1994.
- [Marlin 04] B. Marlin and R. S. Zemel. The multiple multiplicative factor model for collaborative filtering. In *Proc. of the 21st Int'l Conf. on Machine Learning*, pp. 576–583, 2004.
- [丸岡 99] 丸岡章, 瀧本英二. オンライン予測. *人工知能学会誌*, Vol. 14, No. 5, pp. 763–770, 1999.
- [McLaughlin 04] M. R. McLaughlin and J. L. Herlocker. A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In *Proc. of the 27th Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 329–336, 2004.
- [McNee 06a] S. M. McNee, J. Riedl, and J. A. Konstan. Accurate is not always good: How accuracy metrics have hurt recommender systems. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, pp. 1097–1101, 2006.
- [McNee 06b] S. M. McNee, J. Riedl, and J. A. Konstan. Making recommendations better: An analytic model for human-recommender interaction. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, pp. 1103–1108, 2006.
- [McSherry 05] D. McSherry. Explanation in recommender systems. *Artificial Intelligence Review*, Vol. 24, pp. 179–197, 2005.
- [McSherry 09] F. McSherry and I. Mironov. Differentially private recommender systems: Building privacy into the netflix prize contenders. In *Proc. of the 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 627–635, 2009.
- [Mehta 08] B. Mehta and W. Nejdl. Attack resistant collaborative filtering. In *Proc. of the 31st Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 75–82, 2008.
- [Melamed 07] D. Melamed, B. Shapira, and Y. Elovici. Marcol: A market-based recommender system. *IEEE Intelligent Systems*, Vol. 22, No. 3, pp. 74–78, 2007.
- [宮本 99] 宮本定明. クラスタ分析入門 — ファジィクラスタリングの理論と応用. 森北出版, 1999.
- [Mobasher 07] B. Mobasher, R. Burke, R. Bhaumik, and J. J. Sandvig. Attacks and remedies

- in collaborative recommendation. *IEEE Intelligent Systems*, Vol. 22, No. 3, pp. 56–63, 2007.
- [Mooney 99] R. J. Mooney and L. Roy. Content-based book recommending using learning for text categorization. In *ACM SIGIR Workshop on Recommender Systems: Algorithms and Evaluation*, 1999.
- [元田 06] 元田浩, 山口高平, 津本周作, 沼尾正行. データマイニングの基礎. オーム社, 2006.
- [MovieLens] Movielens. University of Minnesota. (<http://movielens.org/>).
- [村上 05] 村上知子, 吉岡信和, 折原良平, 古川康一. Cam 法を用いた個人嗜好モデルに基づく商品推薦システム. *人工知能学会論文誌*, Vol. 20, No. 5, pp. 346–355, 2005.
- [村上 07] 村上知子, 森紘一郎, 折原良平. 推薦結果の意外性を評価する指標の提案. *人工知能学会全国大会 (第 21 回) 論文集*, 2C5-2, 2007.
- [中島 99] 中島義明ほか (編). *心理学辞典*. 有斐閣, 1999.
- [中森 00] 中森義輝. 感性データ解析 — 感性情報処理のためのファジィ数量分析手法. 森北出版, 2000.
- [Nakamura 98] A. Nakamura and N. Abe. Collaborative filtering using weighted majority prediction algorithms. In *Proc. of the 15th Int'l Conf. on Machine Learning*, pp. 395–403, 1998.
- [NetflixPrize 06] Netflix prize, 2006. (<http://www.netflixprize.com/>).
- [Nichols 97] D. M. Nichols. Implicit rating and filtering. In *Proc. of the 5th DELOS Workshop on Filtering and Collaborative Filtering*, pp. 31–36, 1997.
- [沼尾 06] 沼尾雅之. プライバシーと AI. *人工知能学会誌*, Vol. 21, No. 5, pp. 593–601, 2006.
- [O'Connor 99] M. O'Connor and J. Herlocker. Clustering items for collaborative filtering. In *ACM SIGIR Workshop on Recommender Systems: Algorithms and Evaluation*, 1999.
- [O'Connor 01] M. O'Connor, D. Cosley, J. A. Konstan, and J. Riedl. PolyLens: A recommender system for groups of users. In *Proc. of the 7th European Conf. on Computer-Supported Cooperative Work*, pp. 199–218, 2001.
- [O'Donovan 05] J. O'Donovan and B. Smyth. Trust in recommender systems. In *Proc. of the 10th Int'l Conf. on Intelligent User Interfaces*, pp. 167–174, 2005.
- [O'Mahony 04] M. O'Mahony, N. Hurley, N. Kushmerick, and G. Silvestre. Collaborative recommendation: A robustness analysis. *ACM Trans. on Internet Technology*, Vol. 4, No. 4, pp. 344–377, 2004.
- [Ono 05] C. Ono, Y. Motomura, and H. Asoh. A study of probabilistic models for integrating collaborative and content-based recommendation. In *Working Note of IJCAI2005: Multidisciplinary Workshop on Advances in Preference Handling*, 2005.
- [小野 06] 小野智弘, 黒川茂莉, 本村陽一, 麻生英樹. 携帯電話ユーザのための映画嗜好モデル構築と映画推薦システム. *人工知能学会研究会資料, SIG-FPAI-A602-07*, 2006.
- [小野 07] 小野智弘, 本村陽一, 麻生英樹. 携帯電話によるレコメンデーションシステムの一般ユーザによる評価実験. *人工知能学会全国大会 (第 21 回) 論文集*, 2C5-1, 2007.
- [Paterek 07] A. Paterek. Improving regularized singular value decomposition for collaborative filtering. In *Proc. of KDDCup and Wrokshop*, 2007.
- [Pavlov 03] D. Y. Pavlov and D. M. Pennock. A maximum entropy approach to collaborative

- filtering in dynamic, sparse, high-dimensional domains. In *Advances in Neural Information Processing Systems 15*, pp. 1465–1472, 2003.
- [Pazzani 99] M. J. Pazzani. A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*, Vol. 13, pp. 393–408, 1999.
- [Pei 01] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proc. of the 17th Int'l Conf. on Data Engineering*, pp. 215–224, 2001.
- [Pennock 00a] D. M. Pennock, E. Horvitz, S. Lawrence, and C. Lee Giles. Collaborative filtering by personality diagnosis: A hybrid memory- and model-based approach. In *Uncertainty in Artificial Intelligence 16*, pp. 473–480, 2000.
- [Pennock 00b] D. M. Pennock, E. Horvitz, and C. Lee Giles. Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. In *Proc. of the 17th National Conf. on Artificial Intelligence*, pp. 729–734, 2000.
- [Polat 03] H. Polat and W. Du. Privacy-preserving collaborative filtering using randomized perturbation techniques. In *Proc. of the 3rd IEEE Int'l Conf. on Data Mining*, pp. 625–628, 2003.
- [Popescul 01] A. Popescul, L. H. Ungar, D. M. Pennock, and S. Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *Uncertainty in Artificial Intelligence 17*, pp. 437–444, 2001.
- [Rafter 05] R. Rafter and B. Smyth. Conversational collaborative recommendation — an experimental analysis. *Artificial Intelligence Review*, Vol. 24, pp. 301–318, 2005.
- [RecSys] ACM recommender systems. (<http://recsys.acm.org/>).
- [RecSysWiki] ACM RecSysWiki. (<http://www.recsyswiki.com/>).
- [Rendle 09] S. Rendle, L. B. Marinho, A. Nanopoulos, and L. Schmidt-Thieme. Learning optimal ranking with tensor factorization for tag recommendation. In *Proc. of the 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 727–735, 2009.
- [Resnick 94] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. GroupLens: An open architecture for collaborative filtering of Netnews. In *Proc. of the Conf. on Computer Supported Cooperative Work*, pp. 175–186, 1994.
- [Resnick 97] P. Resnick and H. R. Varian. Recommender systems. *Communications of the ACM*, Vol. 40, No. 3, pp. 56–58, 1997.
- [Resnick 11] P. Resnick, J. Konstan, and A. Jameson. Panel on the filter bubble. The 5th ACM Conf. on Recommender Systems, 2011. (<http://acmrecsys.wordpress.com/2011/10/25/panel-on-the-filter-bubble/>).
- [Ricci 07] F. Ricci and Q. N. Nguyen. Acquiring and revising preferences in a critique-based mobile recommender system. *IEEE Intelligent Systems*, Vol. 22, No. 3, pp. 22–29, 2007.
- [Robertson 77] S. E. Robertson. The probability ranking principle in IR. *J. of Documentation*, Vol. 33, No. 4, pp. 294–304, 1977.
- [Rubens 07] N. Rubens and M. Sugiyama. Influence-based active learning for collaborative

- filtering. 人工知能学会研究会資料, SIG-FPAI-A702-01, 2007.
- [齋藤 06] 齋藤堯幸, 宿久洋. 関連性データの解析法 — 多次元尺度構成法とクラスター分析法. 共立出版, 2006.
- [佐久間 09] 佐久間淳, 小林重信. プライバシ保護データマイニング. 人工知能学会誌, Vol. 24, No. 2, pp. 283–294, 2009.
- [Salakhutdinov 07] R. Salakhutdinov, A. Mnih, and G. Hinton. Restricted boltzmann machines for collaborative filtering. In *Proc. of the 24th Int'l Conf. on Machine Learning*, pp. 791–798, 2007.
- [Sarwar 01] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proc. of the 10th Int'l Conf. on World Wide Web*, pp. 285–295, 2001.
- [佐藤 85] 佐藤信. 統計的官能検査. 日科技連, 1985.
- [Seko 11] S. Seko, T. Yagi M. Motegi, and S. Muto. Group recommendation using feature space representing behavioral tendency and power balance among members. In *Proc. of the 5th ACM conference on Recommender systems*, pp. 101–108, 2011.
- [Senecal 04] S. Senecal and J. Nantel. The influence of online product recommendation on consumers' online choices. *Journal of Retailing*, Vol. 80, pp. 159–169, 2004.
- [Shani 02] G. Shani, R. I. Brafman, and D. Heckerman. An mdp-based recommender system. In *Uncertainty in Artificial Intelligence 18*, pp. 453–460, 2002.
- [Shani 05] G. Shani, D. Heckerman, and R. I. Brafman. An mdp-based recommender system. *Journal of Machine Learning Research*, Vol. 6, pp. 1265–1295, 2005.
- [Shani 10] G. Shani. Evaluating recommendation. The 4th ACM Conf. on Recommender Systems, Tutorial, 2010.
- [Shardanand 95] U. Shardanand and P. Maes. Social information filtering: Algorithms for automating “word of mouth”. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, pp. 210–217, 1995.
- [繁榎 06] 繁榎算男, 本村陽一, 植野真臣. ベイジアンネットワーク概説. 培風館, 2006.
- [清水 07] 清水拓也, 土方嘉徳, 西田正吾. 発見性を考慮した協調フィルタリングアルゴリズムにおける評価実験. 電子情報通信学会技術研究報告, WI2 2007–50, 2007.
- [新保 06] 新保史生. フィルタリングと法. 情報の科学と技術, Vol. 56, No. 10, pp. 475–481, 2006.
- [Sinha 02] R. Sinha and K. Swearingen. The role of transparency in recommender systems. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, pp. 830–831, 2002.
- [Smyth 03] B. Smyth and L. McGinty. The power of suggestion. In *Proc. of the 18th Int'l Joint Conf. on Artificial Intelligence*, pp. 127–132, 2003.
- [Smyth 04] B. Smyth, L. McGinty, J. Reilly, and K. McCarthy. Compound critiques for conversational recommender systems. In *Proc. of the IEEE/WIC/ACM Int'l Conf. on Web Intelligence*, pp. 145–151, 2004.
- [Sparling 11] E. I. Sparling and S. Sen. Rating: How difficult is it? In *Proc. of the 5th ACM Conf. on Recommender Systems*, pp. 149–156, 2011.
- [Stevens 51] S. S. Stevens. Mathematics, measurement, and psychophysics. In S. S. Stevens,

- editor, *Handbook of Experimental Psychology*. John Wiley & Sons, 1951.
- [Sundar 08] S. S. Sundar, A. Oeldorf-Hirsch, and Q. Xu. The bandwagon effect of collaborative filtering technology. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, 2008.
- [Sutton 98] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- [Sutton 00] R. S. Sutton and A. G. Barto. 強化学習. 森北出版, 2000. [翻訳] 三上 貞芳, 皆川 雅章.
- [Swearingen 01] K. Swearingen and R. Sinha. Beyond algorithms: An hci perspective on recommender systems. In *SIGIR Workshop on Recommender Systems*, 2001.
- [Sweeney 00] L. Sweeney. Simple demographics often identify people uniquely. Data Privacy Working Paper 3, Carnegie Mellon University, 2000.
- [高間 07] 高間康史, 難波広樹, 岩瀬徳宏, 服部俊一, 武藤優樹, 庄司俊寛. テレビ視聴時の情報推薦に基づくヒューマン・ロボットコミュニケーション. 人工知能学会全国大会 (第 21 回) 論文集, 2D5-5, 2007.
- [竹内 89] 竹内啓 (編). 統計学辞典. 東洋経済新報社, 1989.
- [Thompson 04] C. A. Thompson, M. H. Göker, and P. Langley. A personalized system for conversational recommendation. *Journal of Artificial Intelligence Research*, Vol. 21, pp. 393–428, 2004.
- [徳永 99] 徳永健伸. 情報検索と言語処理, 言語と計算, 第 5 巻. 東京大学出版会, 1999.
- [鷺尾 98] 鷺尾隆, 元田浩. 尺度の理論. 日本ファジィ学会誌, Vol. 10, No. 3, pp. 401–413, 1998.
- [渡辺 05] 渡辺澄夫, 萩原克幸, 赤穂昭太郎, 本村陽一, 福水健次, 岡田真人, 青柳美輝. 学習システムの理論と実現. 森北出版, 2005.
- [Weigend 03] A. S. Weigend. Analyzing customer behavior at Amazon.com. In *The 9th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, Kyenote*, p. 5, 2003.
- [Xue 05] G.-R. Xue, C. Lin, Q. Yang, W.-S. Xi, H.-J. Zeng, Y. Yu, and Z. Chen. Scalable collaborative filtering using cluster-based smoothing. In *Proc. of the 28th Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 114–121, 2005.
- [山本 06] 山本浩司, 片上大輔, 新田克己, 相場亮, 桑田仁. 地図上の情報推薦システムにおける投稿情報の信頼度. 人工知能学会論文誌, Vol. 21, No. 3, pp. 276–286, 2006.
- [山根 06] 山根康男, 官上大輔, 河合由起子, 津田宏, 田中克己. グループ適応型システムのための満足度を考慮した推薦方式の提案. 第 17 回データ工学ワークショップ (DEWS2005), 1A-i7, 2006.
- [山西 03] 山西健司. データとテキストのマイニング. 言語と心理の統計 — ことばと行動の確率モデルによる分析, 統計科学のフロンティア, 第 10 巻, 第 4 章, pp. 179–242. 岩波書店, 2003.
- [Yao 86] A. C. Yao. How to generate and exchange secrets. In *Proc. of Annual IEEE Sympo. on Foundations of Computer Science*, pp. 162–167, 1986.
- [吉井 06] 吉井和佳, 後藤真孝, 駒谷和範, 尾形哲也, 奥乃博. ユーザの評価と音響的特徴との確率的統合に基づくハイブリッド型楽曲推薦システム. 情報処理学会研究報告, 2006 MUS-66, 2006.

- [吉高 07] 吉高淳夫, 脇山孝貴, 平嶋宗. 絵画を例とした鑑賞活動の共有に基づく情報推薦. 人工知能学会全国大会 (第 21 回) 論文集, 2B4-8, 2007.
- [Yu 03] K. Yu, A. Schwaighofer, V. Tresp, W.-Y. Ma, and H.-J. Zhang. Collaborative ensemble learning: Combining collaborative and content-based information filtering via hierarchical bayes. In *Uncertainty in Artificial Intelligence 19*, pp. 616–623, 2003.
- [Zhang 06a] S. Zhang, A. Chakrabarti, J. Ford, and F. Makedon. Attack detection in time series for recommender systems. In *Proc. of the 12th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 809–814, 2006.
- [Zhang 06b] S. Zhang, Y. Ouyang, J. Ford, and F. Makedon. Analysis of a low-dimensional linear model under recommendation attacks. In *Proc. of the 29th Annual ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 517–524, 2006.
- [Ziegler 05] C. N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. In *Proc. of the 14th Int'l Conf. on World Wide Web*, pp. 22–32, 2005.

索引

あ

アイテム間型メモリベース法	53, 59
あてはめ	64
暗黙的評価	31
一時的個人化	6, 59
因子分析	65
永続的個人化	6
オンライン学習	67

か

回帰	64
学習率	18
確信度	47
格付け法	27
確率的潜在意味分析	☞ pLSA
隠れ変数	☞ 潜在変数
可視変数	☞ 観測変数
活動利用者	23
観測変数	71
強化学習	78
協調フィルタリング	34
強度	47
行列分解	65
組み合わせ販売	11
クラス分類	66, 84
グラフィカルモデル	69, 71
グループ推薦	113
効用最適化	11
コールドスタート問題	☞ スタートアップ問題
個人属性情報	33

さ

再現率	15
最大エントロピーモデル	76, 80
採点法	27
サクラ攻撃	101
サクラボット	101
時系列	75
嗜好データ	23, 25
社会的フィルタリング	☞ 協調フィルタリング
順位法	31
情報過多	3

事例ベース法	☞ メモリベース法
推薦システム	2
推薦理由	46
スタートアップ問題	39, 63
正解率	15
精度	15
セレンディピティ	16
潜在的ディリクレ配分法	☞ LDA
潜在変数	70
疎	19
素性関数	76

た

多様性	16, 17, 38, 46
逐次型二項関係学習法	66
定期購入	79
適合アイテム発見	10, 68
適合アイテム列挙	11
デフォルト投票	58
透明性	46
ドメイン知識	38

な

内容ベースフィルタリング	34
能動学習	108

は

灰色の羊問題	63, 70
ハイブリッド推薦システム	86
ハザード関数	79
非個人化	6
被覆率	17, 40
秘密関数計算	97
評価値予測	10
標本利用者	36
不可能性定理	61
プライバシー保護協調フィルタリング	96
プライバシー保護データマイニング	97
ブラックボックス	47
平均絶対誤差	15
ベイジアンネットワーク	69

ホワイトボックス	47	demographic information	33
ま		diversity	17, 38, 46
マルコフ過程	77	domain knowledge	38
マルコフ決定過程	77	dviversity	16
明示的評価	26	E	
メモリベース法	50, 53	EM algorithm	65, 72
モデルベース法	51, 62	EM アルゴリズム	65, 72
や		ephemeral personalization	6, 59
予測精度	14	explanation of recommendations	46
ら		explicit rating	26
ランダム化	99	F	
ランプアップ問題	☞ スタートアップ問題	factor analysis	65
利用者間型メモリベース法	53	feature function	76
利用者データベース	36, 51	finding all good items	11
利用者プロフィール	36	finding some good items	10, 68
利用者 DB	☞ 利用者データベース	fitting	64
わ		folding-in	73
話題多様化	46	G	
A		graphical model	69, 71
accuracy	14, 15	gray sheep problem	63, 70
active learning	108	group recommendation	113
active user	23	GroupLens method	3
B		GroupLens 法	3
Bayesian network	69	H	
blackbox	47	hazard function	79
C		hidden variable	☞ latent variable
classification	66, 84	Horting	65
cold-start problem	☞ start-up problem	human-recommender interaction	19
collaborative filtering	34	hybrid recommender system	86
confidence	47	I	
content-based filtering	34	implicit rating	31
coverage	17, 40	impossibility theorem	61
Cox proportional hazards model	79	information overload	3
Cox 比例ハザードモデル	79	instance-based method	☞ memory-based method
Cross-G-Learn-Relation method	66	item churn	60
cross-selling	11	item-item memory-based method	53, 59
D		L	
decoupled model	74	latent Dirichlet allocation	☞ LDA
default voting	58	latent variable	70
		LDA	74
		learning rate	18

M

Markov decision process	77
Markov process	77
mass customization	4
matrix decomposition	65
matrix factorization	☞ matrix decomposition
Maxent model	☞ maximum entropy model
MaxEnt モデル	☞ 最大エントロピーモデル
maximum entropy model	76, 80
Mean Absolute Error	15
memory-based method	50, 53
model-based method	51, 62

N

no personalization	6
--------------------	---

O

O-I-P モデル	22
observed variable	71
online learning	67
optimizing utility	11
output-input-process model	22

P

persistent personalization	6
personality diagnosis method	81
pLSA	70
precision	15
predicting ratings	10
preference data	23, 25
preferred ordering model	74
privacy-preserving collaborative filtering	96
privacy-preserving data mining	97
probabilistic latent semantic analysis	☞ pLSA

R

ramp-up problem	☞ start-up problem
randomization	99
RankBoost	68
ranking method	31
rating method	27
recall	15
recommender system	2
regression	64
reinforcement learning	78

S

sample user	36
scoring method	27
secure function evaluation	97

secure multiparty computation	97
serendipity	16
shilling attack	101
shilling bot	101
social filtering	☞ collaborative filtering
sparse	19
start-up problem	39, 63
strength	47
subscription	79

T

time series	75
topic diversification	46
transparency	46

U

user database	36, 51
user DB	☞ user database
user profile	36
user-user memory-based method	53

V

visible variable	☞ observed variable
------------------	---------------------

W

whitebox	47
----------	----