

潜在トピックを利用した協調フィルタリングにおける トピック情報源の違いに関する調査

西村 章宏¹ 土方 嘉徳^{2,a)} 酒田 信親¹

受付日 2016年8月14日, 採録日 2017年2月9日

概要: 情報推薦の一手法として協調フィルタリング (CF) がよく知られているが, この CF の中でも Matrix Factorization (MF) は, 評価値の欠損が多い現実のデータセットに対して優れた結果を出していることで有名である. ただし, MF は評価値の欠損に強いものの, 評価値が極端に少ない場合には適用することが困難である. この問題に対して, 評価値だけでなくテキスト情報も利用した Collaborative Topic Regression (CTR) というモデルが近年注目されている. CTR では, トピックを抽出する情報源として, アイテムに関するテキスト情報を用いている. しかし, ユーザに関するテキスト情報を用いた場合の評価は行われてこなかった. 本研究では CTR において, アイテムに関するテキスト情報を用いた場合と, ユーザに関するテキスト情報を用いた場合で, 推薦の性能にどれだけの違いがあるのかを調査する. この両者とベースラインである MF に対して推薦結果の正確性 (再現率) と利便性 (被覆率・多様性) に関して定量的な比較を行う. 実験の結果, ユーザに関するテキスト情報を用いた場合は正確性の高い推薦が実現でき, アイテムに関するテキスト情報を用いた場合は利便性の高い推薦が実現できることが分かった.

キーワード: 推薦システム, 協調フィルタリング, Matrix Factorization, トピックモデル, LDA

Examination on Recommendation Performance of Information Source on Collaborative Filtering Using Latent Topics

AKIHIRO NISHIMURA¹ YOSHINORI HIJIKATA^{2,a)} NOBUCHIKA SAKATA¹

Received: August 14, 2016, Accepted: February 9, 2017

Abstract: Collaborative filtering is one of the popular methods for selecting contents or items suited to user's interest or preference from a content (item) set. Among the existing methods of CF, Matrix Factorization (MF) is superior for the dataset with much missing data. However, even MF cannot perform well when the number of users' ratings are extremely small. To solve this problem, an approach that uses not only users' rating values but also text information of items are becoming popular recently. We focus on one major model of this approach, Collaborative Topic Regression (CTR). In CTR, text information of items is used as information source to extract topics. However, the case using text information of users has not been evaluated. In this study, we compare the case using text information of items and the case using text information of users in CTR to know the performance difference. We compare the both cases and MF (which is a baseline) according to accuracy metrics and usefulness metrics. We found that using text information of users realizes the recommendation with high accuracy and using text information of items realizes the recommendation with high usefulness from the experiment.

Keywords: recommender system, collaborative filtering, Matrix Factorization, topic model, LDA

¹ 大阪大学大学院基礎工学研究科
Graduate School of Engineering Science, Osaka University,
Toyonaka, Osaka 560-8531, Japan

² 関西学院大学商学部
School of Business Administration, Kwansai Gakuin University,
Nishinomiya, Hyogo 662-8501, Japan

1. はじめに

ユーザの情報獲得を支援する方法の1つに推薦システムがある. 推薦システムに関する研究は広く行われており,

^{a)} contact@soc-research.org

それを実現する方法は、ユーザの興味・嗜好に関する情報とアイテムの内容に関する情報をもとに推薦を行う内容ベースフィルタリング (content-based filtering, CBF) と、対象ユーザと嗜好の似ている他者が好むアイテムを推薦する協調フィルタリング (collaborative filtering, CF) に大きく分けることができる [1]。このうち CF は、ユーザやアイテムの内容に関する特徴量を作成する必要がないという簡便さから、多くの商用推薦システム*1で用いられている。CF ではユーザがアイテムを好む程度 (評価値) が行列形式で表現される。この行列は評価値行列と呼ばれる。ただし、実システムやサービスで得られる評価値行列には欠損値が多く存在しており、評価値不足により推薦がうまく行えないことがある (スパース性の問題)。そのため、より少ない評価値から効率的に未知の評価値を推定する手法として、Matrix Factorization (MF) [2], [3] が存在する。

MF はユーザとアイテムをより低次元な潜在特徴空間へマッピングする手法であり、このユーザとアイテムの潜在因子ベクトルを用いて元の評価値を近似することができる。このため、評価値行列中に欠損値が含まれていたとしても、より欠損値の少ない評価値行列を推定することができ、実データに対しても優れた推薦を行うことができる [4]。ただし、MF による欠損値への対処にも限界がある。評価値が極端に少ない新規アイテムや新規ユーザに対してはうまく適用することができない。これは一般にコールドスタート問題と呼ばれる [5], [6]。

この問題に対して、近年の MF の研究では、評価値行列だけでなくアイテムに関するテキスト情報も利用している [7], [8], [9], [10]。これらの研究と MF との大きな違いは、アイテムに関するテキスト情報から潜在トピック (以降は単にトピックと呼ぶ) を抽出し、ユーザとアイテムを表現する潜在因子ベクトルにこのトピックを反映させる点である。抽出したトピックに誘起して潜在因子ベクトルが生成されるため、評価値がまったく存在しない新規アイテムに対しても、評価値が存在する他のアイテムとのトピックの類似関係をもとに評価値を推定することができる [7]。

本研究では、このテキストから抽出したトピックを利用するアプローチの先駆けとなったモデルである Collaborative Topic Regression (CTR) [7] に注目する。CTR では、トピックを抽出する情報源として、アイテムに関するテキスト情報を用いている。一方で、ユーザに関するテキスト情報を用いることに関しては深く言及されていない。ユーザに関するテキスト情報から抽出されるトピックは、主にユーザの種類 (ユーザ層) であると考えられる [11]。アイテムに関するテキスト情報から抽出したトピックを用いると、ユーザが評価したアイテムの特徴を用いて、間接的に対象ユーザの興味を潜在特徴空間へ射影することになる。

一方、ユーザに関するテキスト情報には、対象ユーザの興味や嗜好が直接に、また簡潔に記載されており (5章の実験で用いる論文・記事に関するデータセットでは、論文や記事よりもユーザのプロフィール情報の方がテキストの量が少ない)、そのようなテキスト情報から抽出したトピックを用いれば、アイテムに対して興味のあるユーザ層を、潜在特徴空間でより簡潔に表現できるかもしれない。対象ユーザが評価したアイテム数が少ない場合でも、ユーザに関するテキスト情報があれば、対象ユーザが属するユーザ層を推定することができ、そのユーザ層が好む代表的なアイテムを選択することで正確性の高い推薦を行えると考えられる。そこで我々は、このユーザ層をユーザとアイテムを表現する潜在因子ベクトルに反映させることを考える。

すなわち、本研究では CTR モデルにおいて、トピック情報源としてアイテムに関するテキスト情報を用いた場合 (iCTR と呼ぶ) と、ユーザに関するテキスト情報を用いた場合 (uCTR と呼ぶ) を比較し、推薦性能がどれだけ異なるのかを調べる。この調査では、この両者とベースラインである MF に対して、推薦結果の正確性 (再現率) と利便性 (被覆率・多様性) の両方の観点から、定量的な比較を行う。この際、極端に評価値が欠損している場合でも有効に推薦が行えるかどうかを確認するため、評価値行列のスパース性の程度 (以下スパース率) を変化させて実験を行う。

本研究の貢献をまとめると以下のとおりである。

- (1) CTR モデルにおいて、アイテムに関するテキスト情報を用いた場合と、ユーザに関するテキスト情報を用いた場合を比較し、情報源の違いが推薦結果に与える影響について分析した。
- (2) 性能評価には正確性に関する評価指標に加えて、推薦の利便性を表す被覆率と多様性に関する評価指標も用いた。
- (3) 上記モデルの性能評価において、スパースさに対するロバスト性について分析した。

本稿では、2章で関連研究を紹介する。3章で本研究が対象とする CTR モデルと、その情報源としてユーザに関する情報を用いた場合のモデルについて説明する。4章で情報源の違いの推薦性能への影響に関する評価方法を、5章で比較実験の詳細とその結果を示す。最後に、6章でまとめを述べる。

2. 関連研究

協調フィルタリング (CF) を用いた推薦手法は大きく2種類に分けられる [12]。1つは、評価値から直接ユーザ間またはアイテム間の比較を行うメモリベース法である。ピアソン相関係数でユーザ間またはアイテムの類似度を測る手法がよく知られている [13], [14]。もう1つは、評価値を利用して推薦を行うためのモデルを学習するモデルベー

*1 Amazon (<http://www.amazon.com/>) や LinkedIn (<https://www.linkedin.com/>), Hulu (<http://www.hulu.com>) など

手法である。評価値を予測する関数のパラメータを回帰分析や因子分析などのアプローチで学習するモデル [2], [15] や、ユーザ・アイテム・評価値をそれぞれ確率変数ととらえて同時分布を学習するモデル [16] などが存在する。なかでも、元の評価値行列を複数の低次元の行列に分解し、ユーザやアイテムが持つ潜在特徴を利用するモデルが高い性能を発揮することが知られている [4], [17]。これらは、行列因子分解モデルと呼ばれる。行列因子分解モデルの中でも、 $R \approx U^T V$ となるように元の評価値行列 R を近似する行列 U, V を推定するモデル [2] は、実装が容易かつ高性能であることから広く利用されており、一般的に Matrix Factorization (MF) と呼ばれている。

このような潜在特徴に関しては、自然言語処理の分野でも活発に研究が行われており、近年ではトピックモデルと呼ばれる手法が注目されている。これは、文書の生成過程を潜在変数を交えて階層的に表現し、各階層間の関係を確率的にモデル化したものである。これにより文書集合から容易にトピック（文書中の潜在因子）を抽出することが可能である。トピックモデルの中で最も基本的なモデルは、Blei らが提案した Latent Dirichlet Allocation (LDA) [18] であり、モデルの階層構造が比較的シンプルでありながら強力なトピック抽出機能を有している。

近年では、CF に評価値以外の情報を利用することで、コールドスタート問題やスパース性の問題に対処しようとする研究が活発に行われている。これらの研究は CF と内容ベースフィルタリングを組み合わせたハイブリッド法 [19] と見なせる。ハイブリッド法には、両方のモデルの結果を混合して提示する方法や、評価値の数に応じてモデルを切り替える方法などが存在する [19]。その中でも、両方のモデルに前後関係がなく、同等のレベルでモデル化を行う方法に注目する。この方法は階層的な確率モデルとして表現することが可能であるため [5], [20], [21]、先述のトピックモデルを自然な形で取り入れることができる。そのため、CF とトピックモデルを組み合わせたアプローチが研究され始めている [7], [8], [9], [10]。特に、collaborative topic regression (CTR) [7] は、Wang と LDA の提案者である Blei が提案したもので、CF とトピックモデルを融合する先駆けとなった手法である。CTR はシンプルなモデルでありながらも、優れた推薦性能を実現している。手法の基本としては、アイテムに関するテキスト情報からトピックを抽出し、それをアイテムの特徴を表現する潜在因子ベクトルに反映させている（詳細は 3 章参照）。

3. 推薦モデル

本研究では CTR モデルにおいて、トピックを抽出する情報源の違いに着目する。具体的には、アイテムに関するテキスト情報からトピックを抽出するモデルと、ユーザに関するテキスト情報からトピックを抽出するモデルが考え

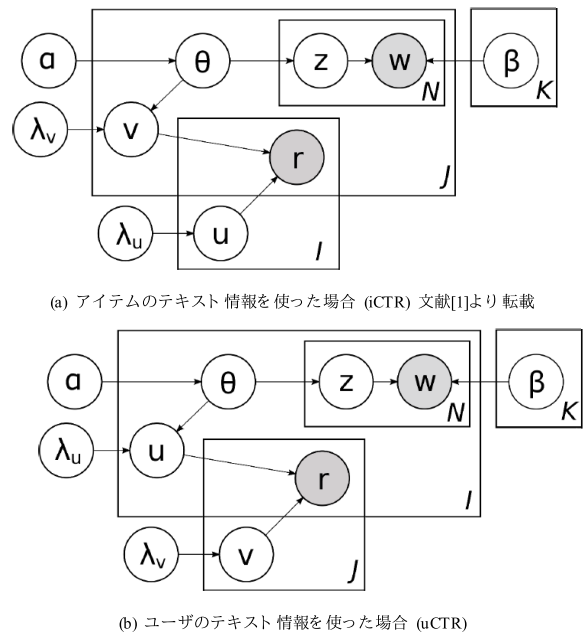


図 1 Collaborative Topic Regression のグラフィカルモデル
Fig. 1 Graphical model of Collaborative Topic Regression.

られる。この章では、これらのモデルの概要を説明する。

3.1 Collaborative Topic Regression (item-oriented)

CTR モデル [7] は、協調フィルタリングにトピックモデルを融合させるアプローチの先駆けとなったモデルである。図 1-(a) に、CTR モデル（本稿では iCTR）のグラフィカルモデルを示す。図の上半分（左端の α から右端の β までの横の並び）はトピックモデルの LDA [18] の生成過程とはほぼ同じであり、下半分 (v や λ_v 以下の部分) は MF に相当する処理を表している。この MF の部分は、確率的モデルである Probabilistic Matrix Factorization (PMF) [22], [23] がベースとなっている。図中の各変数については以下に続く節で詳しく説明するが、ポイントのみ押さえて説明すると、CTR は LDA 部分の θ と PMF 部分の v との間に依存関係を与えたモデルだということが分かる。この依存関係により、アイテムに関する文書から抽出したアイテムを特徴づけるトピック比率 θ をアイテムの潜在因子 v へ反映させている。

3.1.1 モデル説明

最初に、このモデルに登場する変数に関して説明を行う。添え字として、各ユーザを i (総数 I)、各アイテムを j (総数 J)、各トピックを k (総数 K) と表す。実際にデータとして得られる観測変数は、アイテムに関するテキスト中の単語（トークン）列 w_j 、ユーザのアイテムに対する評価値 r_{ij} である。今回行うタスクにおいて、評価値は $r_{ij} \in \{0, 1\}$ の二値である。 $r_{ij} = 0$ は欠損値であり、ユーザ i がアイテム j に興味を持っていない、もしくは存在を知らないこ

とを意味する。また、全アイテムの w_j を集め、その中でユニークな単語の集合（語彙集合）を定義し、各語彙を表す添字を y （総数 Y ）とする。加えて、 w_j の各トークンを指す添え字を n とする。潜在変数である z_{jn} は、トークン w_{jn} に対して割り当てられるトピックである。

これらの観測変数と潜在変数のほかに、モデルを構成する要素としてパラメータが存在する。パラメータは確率変数の生成確率を表現する変数であり、これを推定することがモデルの学習であるといえる。トピック比率 θ_j は $\sum_k \theta_{jk} = 1$ を満たし、各アイテムが持つトピックの傾向を表している。この θ_j の比率に基づき、多項分布から確率的に z_{jn} が生成される。語彙比率 β_k は $\sum_y \beta_{ky} = 1$ を満たし、トピック k における語彙の出現確率を表している。 u_i はユーザーの特徴を表現する K 次元の潜在因子ベクトル、 v_j はアイテムの特徴を表現する K 次元の潜在因子ベクトルである。

パラメータの分布を制御する変数として、ハイパーパラメータが存在する。 λ_u, λ_v は、それぞれ u_i, v_j を生成する正規分布の分散を制御するハイパーパラメータであり、各 u_i, v_j に対して平滑化（要素値が極端な値をとることの抑制）を行う効果がある。 α は、 θ_j を生成するディリクレ分布のハイパーパラメータであり、アイテム全体における生成されるトピックの偏り具合を制御する。

MF を確率モデルとして一般化した PMF [22], [23] では、評価値は式 (1) により生成される。

$$\begin{aligned} u_i &\sim N(0, \lambda_u^{-1} I_K) \\ v_j &\sim N(0, \lambda_v^{-1} I_K) \\ r_{ij} &\sim N(u_i^T v_j, c_{ij}^{-1}) \end{aligned} \quad (1)$$

ただし、 I_K は K 次元の単位行列、 $N(\mu, \sigma)$ は正規分布を表す。式中の c_{ij} は、既知の評価値 r_{ij} の影響を制御するハイパーパラメータである。欠損値となる要素に対しては、該当する要素を $c_{ij} = 0$ と設定することでその影響を無視する。特に $c_{ij} = 1$ の場合、PMF の最大事後確率 (MAP) 推定は、MF の目的関数を最小化 (式 (2)) した結果と一致する。

$$\min_{U, V} \sum_{ij} (r_{ij} - u_i^T v_j)^2 + \lambda_u \|u_i\|^2 + \lambda_v \|v_j\|^2 \quad (2)$$

ただし、第 2 項と第 3 項は正則化項である。

CTR では、 c_{ij} は次のように定義する。

$$c_{ij} = \begin{cases} a & (r_{ij} = 1 \text{ のとき}) \\ b & (r_{ij} = 0 \text{ のとき}) \end{cases} \quad (3)$$

ここで a と b はモデルのチューニング用のパラメータである。既知の評価値 r_{ij} の影響を制御するためのもので、既知の場合より強くモデルに反映させるため、 $a > b > 0$ を満たす必要がある。

次に、CTR の生成過程を説明する。

1. 各ユーザー i において、正規分布 $N(0, \lambda_u^{-1} I_K)$ から u_i を生成。
2. 各アイテム j において、
 - a. デイリクレ分布 $Dir(\alpha)$ から θ_j を生成。
 - b. 正規分布 $N(0, \lambda_v^{-1} I_K)$ から K 次元ベクトル ϵ_j を生成し、 $v_j = \epsilon_j + \theta_j$ とする。
 - c. 各トークン w_{jn} において、
 - i. 多項分布 $Mult(\theta)$ から、トピック z_{jn} を生成。
 - ii. 多項分布 $Mult(\beta_{z_{jn}})$ から、トークン w_{jn} を生成。
3. 各ユーザー・アイテムのペア (i, j) において、正規分布 $N(u_i^T v_j, c_{ij}^{-1})$ から r_{ij} を生成。

3.1.2 モデルの学習

この CTR モデルのパラメータ (u, v, θ, β) の学習は、元論文 [7] に従いモデルの事後確率 L (式 (4)) を最大化するように反復的に最適化を行う。各パラメータの更新は、更新するパラメータ以外を固定して事後確率 L を最大化し、更新したいパラメータを最適化する coordinate ascent 法 [24] により行う。

$$\begin{aligned} L = & - \sum_i \sum_j c_{ij} (r_{ij} - u_i^T v_j)^2 \\ & + \sum_j \sum_n \log \left(\sum_k \theta_{jk} \beta_{k, w_{jn}} \right) \\ & - \lambda_u \sum_i \|u_i\|^2 - \lambda_v \sum_j \|v_j - \theta_j\|^2 \end{aligned} \quad (4)$$

u, v の更新は、式 (5) により行う。式中の U, V はそれぞれ列ベクトル u_i, v_j を並べた行列、 C^i, C^j は対角行列でそれぞれ $C_{dd}^i = c_{id}, C_{dd}^j = c_{dj}, r_i, r_j$ はベクトルでそれぞれ $r_i = (r_{i1}, r_{i2}, \dots, r_{iJ}), r_j = (r_{1j}, r_{2j}, \dots, r_{Ij})$ である。各 u_i に対して事後確率 L を大きくする勾配方向へ u_i を微量ずつ変化させていく。各 v_j に対しても同様に変化させていく。

$$\begin{aligned} u_i &= (VC^i V^T + \lambda_u I_K)^{-1} VC^i r_i \\ v_j &= (UC^j U^T + \lambda_v I_K)^{-1} (UC^j r_j + \lambda_v \theta_j) \end{aligned} \quad (5)$$

θ の更新において、事後確率 L (式 (4)) の第 2 項目を直接求めることが非常に困難である。そこで、式 (4) から θ_j を含む項を取り出し、アイテムごとに分離して $L(\theta_j)$ とする。次に、 $\phi_{jnk} = q(z_{jn} = k)$ を定義し、Jensen の不等式を利用して式 (6) のように変形する。

$$\begin{aligned} L(\theta_j) &= -\lambda_v \|v_j - \theta_j\|^2 + \sum_n \log \left(\sum_k \frac{\theta_{jk} \beta_{k, w_{jn}} \phi_{jnk}}{\phi_{jnk}} \right) \\ &\geq -\lambda_v \|v_j - \theta_j\|^2 \\ &\quad + \sum_n \sum_k \phi_{jnk} (\log(\theta_{jk} \beta_{k, w_{jn}}) - \log(\phi_{jnk})) \\ &= L(\theta_j, \phi_j) \end{aligned} \quad (6)$$

この $L(\theta_j, \phi_j)$ は、本来求めたい $L(\theta_j)$ の下限となって

いる。また、最適な ϕ_{jnk} は $\phi_{jnk} \propto \theta_{jk} \beta_{kw_{jn}}$ を満たす。 θ_j の最適化に関しては、解析的に行うことができないため、 θ_j が simplex であることを制約条件に projection gradient 法 [24] を利用して探索的に最適化を行う。

最後に β の更新は、式 (7) により行う。

$$\beta_{kw} \propto \sum_j \sum_n \phi_{jnk} \delta(j, n, w) \quad (7)$$

$$\delta(i, n, w) = \begin{cases} 1 & (w_{jn} = w \text{ のとき}) \\ 0 & (w_{jn} \neq w \text{ のとき}) \end{cases}$$

以上のパラメータ (u, v, θ, β) の更新処理を、事後確率の変化が微小になり収束したと見なせるまで繰り返し実行する。

3.1.3 評価値の予測と推薦

モデルの学習が完了した後、推定したパラメータ \hat{u}, \hat{v} を用いて式 (8) から予測評価値 \hat{r} を計算することができる。この予測評価値は実数値となるため、値の大きい順にランキングを作成することができる。推薦時には、あるユーザー i に関連する評価値 r において、値が 0 の要素 (i, j) を抽出し、それらの予測評価値 \hat{r} を求める。そして、予測評価値 \hat{r}_{ij} の値が大きい順にアイテム j を列挙したものを、推薦リストとしてユーザーへ提示する。

$$r_{ij} \approx \hat{r}_{ij} = u_i^T v_j \quad (8)$$

3.2 Collaborative Topic Regression (user-oriented)

前節の CTR モデルでは、トピックをアイテムに関する情報源から抽出していたが、この情報源をユーザーに関する情報源に変更した場合について考える。すなわち、図 1(a) に示したグラフィカルモデルにおいて、 u と v および関連する $\lambda_u, \lambda_v, I, J$ を入れ替えたモデル (uCTR) を構築する。uCTR のグラフィカルモデルを図 1(b) に示す。uCTR と iCTR の違いは、トピックを抽出する情報源が異なる点である。このため、uCTR モデルでは抽出される K 次元のトピック分布 θ の内容もユーザーに関するトピックとなり、このトピックの影響を受ける潜在因子ベクトル u, v はユーザーに関する潜在因子となる。uCTR のパラメータ学習に関しては、3.1.2 項と同様に最大事後確率 (MAP) 推定により行う。

4. 評価方法

推薦結果の正確性と利便性を評価する。交差検定により評価指標の値を算出して定量的に評価を行う。

4.1 評価方法の概要

推薦システムの性能を調べるため、推薦の正確性を表す指標として再現率、推薦の利便性を表す指標として被覆率と多様性の評価を行う。本評価では不特定多数のユーザーに

よる評価データを利用するため、交差検定によるオフライン評価 [25] を行う。具体的には、データを 5 分割して訓練データ 4 つとテストデータ 1 つに分け、テストデータを擬似的にユーザーが好む未知のアイテム (正解データ) と見なす。訓練データを利用してモデルの学習を行った後、上位 N 個の推薦アイテムを推薦リストとして獲得する。推薦リストと正解データを比較して評価指標の値を算出する。

4.1.1 評価指標

正確性と利便性について、本研究で利用する評価指標を紹介する。

正確性に関する評価としては、適合率 (Precision) と再現率 (Recall) が代表的であるが、元論文 [7] や類似の研究 [8], [9], [23] に倣い、再現率を利用する。インターネット上には非常に多くのコンテンツ (アイテム) が存在しているが、これらのアイテムをユーザーが網羅的に閲覧することは不可能である。そのためオフライン評価では、ユーザーがまだ閲覧していないが閲覧すれば好むと予測されるアイテムが「好きでない」と扱われる。上述の研究では、この問題による影響を抑えるため、正確性の評価に適合率ではなく再現率を採用している。

また、利便性の評価指標としては、推薦システムが偏りなく様々なアイテムを実際に推薦できているかを表す被覆率と、ユーザーごとに個別化された推薦ができているかを表す多様性を採用する。被覆率は、全アイテムの中で実際に推薦可能なアイテム (テストセット中で推薦されたアイテム) がどの程度存在するかを表したものである。被覆率には大きく分けると、Prediction Coverage (PC) と Catalogue Coverage (CC) [26] の 2 種類がある [27] が、推薦結果のみから計算可能な Catalogue Coverage (CC) を用いる。多様性を表す指標には、凝集多様性 (aggregate diversity) [28], ユーザー間相違度 (inter-user diversity) [29], 時間的多様性 (temporal diversity) [30] などの指標が存在する [25] が、本研究では個々のユーザーごとに多様な推薦結果が得られているかを直接的に示すユーザー間相違度 (IUD) を用いる。

あるユーザー i に提示される推薦リストを L_i としたときの、それぞれの評価指標は以下のように算出される。なお、 \mathcal{S} は、リストからのリスト内のアイテム集合への写像を表す。

再現率 (Recall)

$$Recall = \frac{|T_i \cap \mathcal{S}L_i|}{|T_i|} \quad (9)$$

ただし、 T_i はテストセット中のユーザー i の好きなアイテム集合である。最終的なシステムの再現率は、全ユーザーの Recall の平均をとることで求める。

被覆率 (CC)

$$CC = \frac{|\cup_{i=1, \dots, I} \mathcal{S}L_i|}{|B|} \quad (10)$$

ただし、 B はテストセット中のアイテム集合である。CC が低いほど、推薦システムは全アイテム中の一部のアイテムしか推薦できないことを示す。

多様性 (IUD)

$$IUD = \frac{1}{|U|C_2} \sum_{u_1}^U \sum_{u_2}^U d_{u_1, u_2} \quad (11)$$

$$d_{u_1, u_2} = 1 - \frac{|\mathcal{S}L_{u_1} \cap \mathcal{S}L_{u_2}|}{S}$$

ただし、 $S = |\mathcal{S}L_{u_1}| = |\mathcal{S}L_{u_2}|$ である。この式より、IUD は任意の2人のユーザ u_1, u_2 に注目したときに、互いの推薦リストに含まれるアイテムがどれほど違っているかを測定していることが分かる。IUD は $[0, 1]$ に正規化されたユーザ間の距離の平均ととらえることができ、0に近いほどユーザ間の推薦結果に差はなく、1に近いほど差が大きいことを示している。

5. 実験

iCTR, uCTR, MF の比較実験を行う。使用するデータセットは、元論文 [7] に倣い、ソーシャルブックマークサービス CiteULike *2におけるユーザがブックマークを行った論文・科学記事である。我々は、CiteULike において2014年9月から11月の間にブックマークを行ったユーザとそのユーザのすべてのブックマーク、およびそのブックマークの対象となったすべての論文・科学記事を、CiteULike を直接クロールすることで収集した。また、ユーザに関してはそのプロフィール情報を、論文・科学記事に関してはそのタイトルと要約情報も収集した。

5.1 前処理

CiteULike で扱っている論文や科学記事の大半は英語で書かれている。そのため、実験でも英語のテキストを対象とする。英語の文章に対しては、正規化(小文字への統一とステミング)を行った後、ストップワード(he, the, this など)を除去することが一般的である。本実験では、ステミングに関しては TreeTagger *3を利用し、ストップワードの除去に関しては SlothLib *4を利用した。

5.2 実験設定

CiteULike のデータでは、ユーザの評価値は記事ブックマーク(ある論文・記事をブックマークしたかどうか)である。そのため、ユーザがあるアイテム(論文・記事)に関心がある場合には1、関心がないもしくは存在を知ら

*2 Web上に存在する論文やWebページのブックマークを行い、それらを整理・共有できるサービス

*3 <http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/>

*4 <http://svn.sourceforge.jp/svnroot/slothlib/CSharp/Version1/SlothLib/NLP/Filter/StopWord/word/English.txt>

表 1 ユーザに関するテキスト情報の例

Table 1 An example of text information on user.

<p>“Profile” の例 (3人分)</p> <ul style="list-style-type: none"> - PhD, ATR specialization in art therapy. - I am a member of the Software Engineering - Productivity tools department at *** - India. - I am generally interested in studies of learning and the design of instruction, especially in the areas of reading comprehension, science, and mathematics. In particular, I am interested in how objects external to the person (physical artifacts, symbolic and graphical representations) affect learning and reasoning.
<p>“Interest” の例 (3人分)</p> <ul style="list-style-type: none"> - Cheminformatics, chemometrics, statistics, chemistry, metabolomics, systems biology, semantic web. - Complex networks, Spatial distribution of networks, Telecommunications, Graph theory, Network design - I work on Geno-transcriptomic study of solid tumours, diffusely infiltrating and thus with a complex architecture. My interest are therefore inter and intra tumour heterogeneity and tumour evolution. My biological background is based on Genomics and few Transcriptomics and Proteomics and Protein Networks. I have a technical biological background in : SNP and Expression Microarrays. FISH, PCR, Western Blot. And in Informatics : R, Python, SQL, PHP, Perl.

ない場合には0となる。iCTRでは、トピック抽出元のテキストにはアイテムの“article title”と“abstract”を用いる。uCTRでは、トピック抽出元のテキストにはユーザの“profile”と“interest”(ユーザが興味のある分野をキーワードで列挙したもの)を用いる。

“profile”と“interest”の例(3人分)を表1に載せる。“profile”と“interest”の単語数の平均(標準偏差)はそれぞれ、39.2(6.25)と15.1(3.88)であった。これから分かるように、ユーザに関するテキスト情報は、アイテム(論文や科学記事)に関するテキスト情報よりも、テキスト量が少ないことが分かる。また、これらのテキストには、そのユーザの興味のある分野を簡潔に表すキーワードが含まれていることが分かる。

また、用いたデータセットの評価値行列のスパース率を計算したところ、99.926%と非常にスパースであった。我々はスパース率による結果の変化を調べるため、被ブックマーク数がある閾値以下のアイテムを除外することでスパース率を低下させる処理を行った。本実験では、被ブックマーク数が2, 4, 6, 8, 10未満のアイテムを除外した際のそれぞれの結果(順に nlt=2, 4, 6, 8, 10)を求め、スパース率の増減によるモデルごとの性能の変化についても分析する。データセットの主な統計量(スパース率と評価値存在率を含む)を表2に示す。

表 2 データセットの統計量
Table 2 Statistical data of our dataset.

	アイテム数	ユーザ数	ユニーク 単語数 (item)	ユニーク 単語数 (user)	総単語数 (item)	総単語数 (user)	スパース率 (%)	評価値 存在率 (%)
nlt=2	51,435	6,671	39,202	1,121	4,779,689	15,160	99.926	0.074
nlt=4	26,575	6,416	25,027	1,091	2,474,700	14,930	99.894	0.106
nlt=6	10,925	5,969	14,354	1,034	1,012,098	14,027	99.827	0.173
nlt=8	5,796	5,597	9,657	1,012	526,410	13,135	99.754	0.246
nlt=10	3,534	5,312	7,364	926	321,581	12,164	99.678	0.324

モデルのハイパーパラメータ設定は、グリッドサーチにより行った。潜在因子ベクトルの次元に相当するトピック数 K については、特徴の表現力がある程度維持しつつ、特徴量の次元数が可能な限り小さい値となるように、段階的に設定して評価指標の値を確認した。具体的には、トピック数 ($K = 5, 10, 30, 50, 100, 200$) において評価指標の値の向上が微小になった $K = 50$ を用いた。潜在因子ベクトルの平滑化度合いを調整する λ_v, λ_u については、元論文 [7] を参考に iCTR では $\lambda_v = 1, 10, 100, \lambda_u = 0.1, 1, 10$, uCTR では λ_v と λ_u の値を入れ替えた設定で、それぞれの組合せを調べた際の最も良い結果 (iCTR では $(\lambda_v = 100, \lambda_u = 0.1)$, uCTR では $(\lambda_v = 0.1, \lambda_u = 100)$) を用いた。残りのハイパーパラメータに関しては、元論文 [7] と同じ値を用いた ($a = 1, b = 0.01, \alpha = 1$)。パラメータ θ と β の初期値に関しては、事前にデータセットに用いる文書集合を使って LDA モデルの学習を行い、その結果得られる θ と β の値を利用する (LDA の学習は Collapsed Gibbs Sampling [31] で 500 回の反復を行った)。その他のパラメータに関しては、乱数により初期化を行う。なお、MF の実装は、元論文 [7] の方法を参考に、CTR モデルのハイパーパラメータを ($\lambda_v = 0.1, \lambda_u = 0.1, a = 1, b = 0.01, \alpha = 1$) と設定し、かつ θ の値をすべて 0 に固定したものを利用する。

5.3 結果・考察

5.3.1 Perplexity

実験結果における各評価指標を見る前に、まずは構築したモデルにおける Perplexity を確認する。iCTR を対象とした場合の Perplexity は式 (12) で表される。

$$Perplexity(D) = \exp \left\{ - \frac{\sum_j \sum_n \log(\sum_k \theta_{jk} \beta_{k, w_{jn}})}{\sum_j \sum_n 1} \right\} \quad (12)$$

この式において、変数は 3.1 節のものと同じである。uCTR を対象とする場合は、 j をユーザと見なせばよい。Perplexity は言語モデルの複雑さを表す指標であり、一般的には小さい方が良くとされる。ただし、今回の推薦タスクにおいては、利用するテキストの違いにより、学習されるモデルの複雑さに違いが生まれるか否かを確認するため

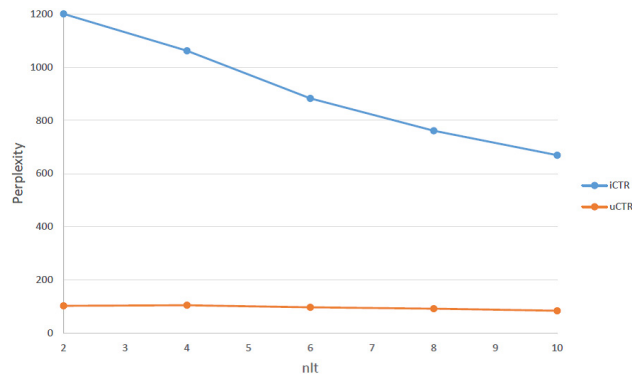


図 2 Perplexity (nlt=2 が最もスパース)
Fig. 2 Perplexity (the most sparse when nlt=2).

に用いる。

今回のデータセットにおける、収束時の各 nlt での Perplexity を図 2 に示す。図 2 の結果を見ると、今回のデータセットでは iCTR よりも uCTR はモデルの複雑さが小さいことが分かる。すなわち、uCTR の方が確率的に選択される単語はより限定的であり、生成されるテキストの曖昧性が小さいことを意味している。また、nlt の値が増加しても、Perplexity の値はほぼ一定であることから、スパース率による影響も uCTR の方が小さいことが分かる。よって、学習されたモデルの性能の面では、iCTR よりも uCTR の方が良いといえる。

ただし、iCTR と比べて uCTR のモデルの複雑さは非常に小さいので、ユーザに関するテキストが持つ情報はアイテムに関するテキストが持つ情報よりも乏しい可能性がある。そのため、Perplexity が低くても推薦結果が必ずしも良くなるとは限らない。以降の項で、推薦の性能について評価する。

5.3.2 再現率

正確性の指標である再現率 (Recall) の結果を図 3 に示す。このグラフは iCTR, uCTR, MF のモデル間の比較、および 5 段階に分けたスパース率間での比較を行っている。横軸は推薦リストに含めるアイテムの数 (推薦数) である。まず同一モデルにおけるスパース率の変化による影響に注目すると、どのモデルにおいてもスパース率が高いほど (nlt の値が小さいほど)、Recall の値は小さくなる傾

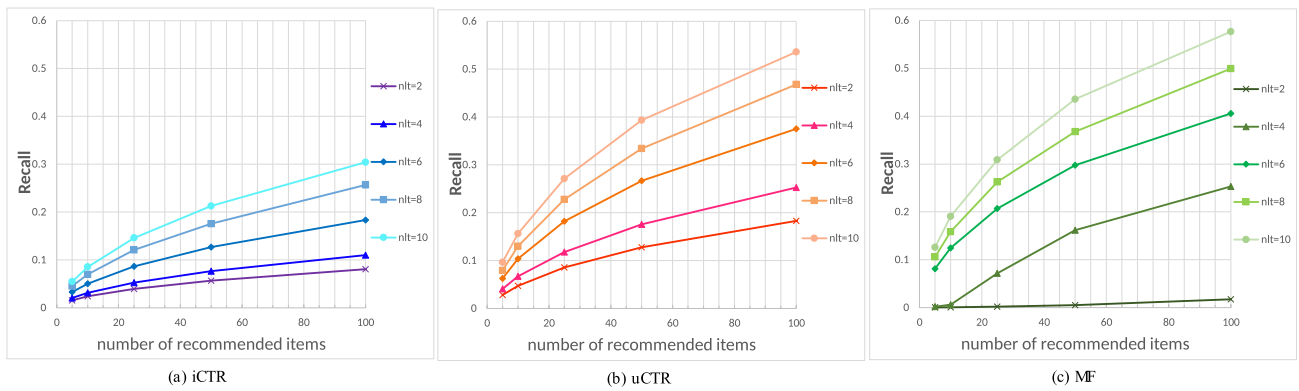


図 3 モデル・スパース率ごとの Recall の結果 (nlt=2 が最もスパース)

Fig. 3 Recall in each model and in each sparse ratio (the most sparse when nlt=2).

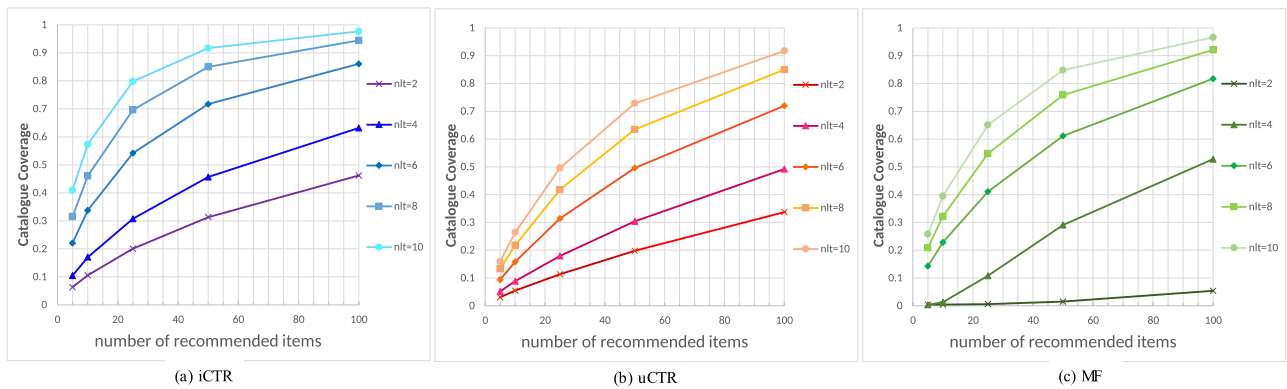


図 4 モデル・スパース率ごとの Catalogue Coverage の結果 (nlt=2 が最もスパース)

Fig. 4 Catalogue Coverage in each model and in each sparse ratio (the most sparse when nlt=2).

向が見られる。

ここで興味深いことに、両 CTR と MF とではスパース率の変化にともなう Recall の変化量が大きく異なっている。推薦数が 100 となる値 (グラフ右端) に注目すると、MF ではスパース率が最大のとき (nlt=2) と最小のとき (nlt=10) の間で Recall の差は約 0.55 となっている。対して、iCTR はスパース率が最大のときと最小のときの間で Recall は約 0.2 しか変わらず、uCTR は約 0.35 しか変わらない。MF に比べるとスパース率による影響が小さく抑えられている。これらのことから、iCTR と uCTR は MF に比べてデータの欠損に対してロバストであることが分かる。

データのスパース率が小さい場合は、MF が最も性能が良くなっている。これは、CTR でテキストから抽出されたトピックの情報よりも、評価値行列に存在する潜在的な情報の方がよりリッチであり、トピックの情報にノイズとなってしまうためと考えられる。iCTR と uCTR を比べると、データのスパース率によらず、uCTR の方が iCTR よりも Recall が高くなっている。この原因を解明することは困難であるが、アイテムのトピックを利用するよりも、ユーザのトピックを利用する方が、推薦の正確性を向上させる可能性があることを示す結果といえる。

以上の結論としては、iCTR、uCTR とともに評価値からの

情報不足をトピックにより補うことができている、スパース率の影響を抑えることができるといえる。特に uCTR は、より推薦の正確性を向上させる可能性があることを確認できた。ただし、評価値情報が十分である場合には、MF が最も正確性が高くなっていることから、トピックの導入がノイズをもたらしている可能性があることに注意する必要がある。

5.3.3 被覆率

被覆率を表す Catalogue Coverage (CC) の結果を図 4 に示す。同一モデルにおけるスパース率の変化による影響は、Recall の傾向と似たものとなった。すなわち、スパース率が高いほど (nlt の値が小さいほど)、CC の値は小さくなる傾向が見られた。スパース率が高い場合と低い場合の CC の値の差は、推薦数が増えるにつれ、大きくなった。

手法間で比較を行うと、iCTR がいずれのスパース率においても、またいずれの推薦数においても、他の 2 手法より優れた結果となっている。MF の結果は uCTR の結果より優れているが、スパース率が非常に低い場合 (nlt=2) では、極端に CC が低くなっており uCTR の CC よりも低くなっている。このことより、トピックの導入はスパース率が高い場合に、CC が極端に減少する現象を抑制する効果があることが分かる。

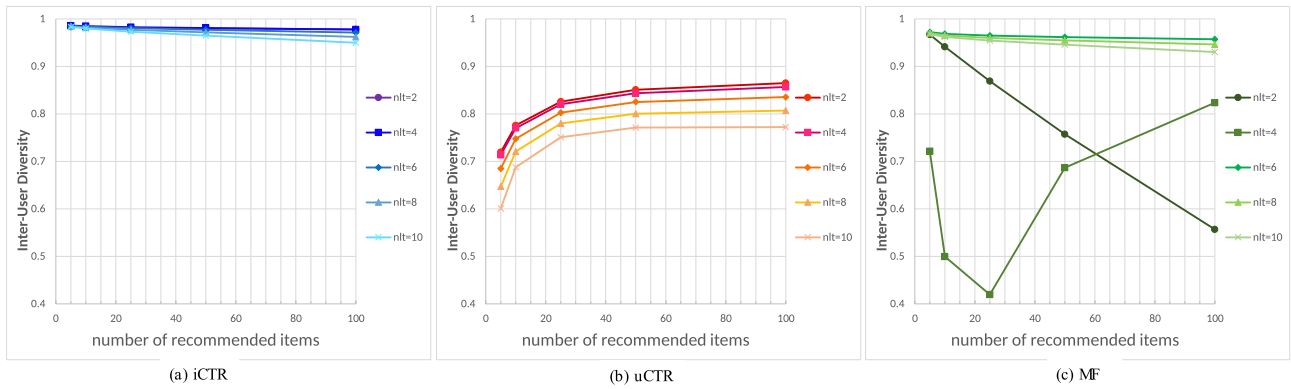


図 5 モデル・スパース率ごとの Inter-User Diversity の結果 (nlt=2 が最もスパース)

Fig. 5 Inter-User Diversity in each model and in each sparse ratio (the most sparse when nlt=2).

以上より、アイテムのトピックを利用すると、推薦結果の被覆率を上げることができることが分かった。それに比べると、ユーザのトピックを導入は、推薦結果の被覆率の向上には寄与しなかった。しかし、MF はスパース率が高い場合に極端に被覆率が低下する現象が見られ、アイテムまたはユーザのトピックを導入すると、その低下の程度を緩和できることが分かった。

5.3.4 多様性

また別の利便性の指標として、多様性を表す Inter-User Diversity (IUD) の結果を図 5 に示す。このグラフを見ると、iCTR では、スパース率を変化させても、またいずれの推薦数でも、IUD は非常に高い値で安定しており、3つの手法の中で最も良い結果となった。したがって、ユーザごとに個別化された多様な推薦が実現できているといえる。

uCTR は、iCTR や MF (スパース率が低い場合) より悪い結果となった。特に、推薦数が 25 未満の場合に着目すると、IUD が 0.6–0.8 程度と低い値となっている。これは、推薦リストの上位において個別化できていないことを示す。推薦リストの上位においてはメジャーなアイテムを推薦し、下位になるとようやく個別化されたアイテムを推薦しているのだと思われる。

MF は、スパース率が低い場合は、iCTR に匹敵するほどの良い結果となった。しかし、一部の結果 (MF (nlt=2) と MF (nlt=4)) がほかとは異なる挙動を示していることが見て取れる。MF (nlt=2) は、急激に単調減少している様子が見られるが、評価値不足のため推薦機構が有効に働かなくなっているためだと考えられる。MF (nlt=4) は、複雑な挙動を示しており、推薦数 25 以降では IUD が上昇する様子が見られる。我々はこの原因を特定することはできなかったが、評価値行列のスパース性が MF の挙動にもたらす複雑さの 1 つとしてとらえる必要がある。以上より、MF ではスパース率が高くなると IUD の値が急激に減少したり不安定になったりする傾向があるが、トピックを導入した iCTR と uCTR は、その現象を起こさないよう

にすることができるといえる。

5.4 結果のまとめ

推薦の正確さを重視するのであれば、テキストから抽出したトピックを利用するのではなく、オリジナルの MF を用いるほうが良い。しかし、データセットのスパース率が高い場合は、uCTR を用いるほうが良い。MF は、スパース率によって大きくその正確性が変わってしまうため、システム運用後のスパース率の値が想定できない場合は、uCTR を用いるほうが無難といえる。

また、システム全体として多くの種類のアイテムが推薦対象となるようにするためには iCTR を用いるほうが良い。特に推薦数が少ない場合には有効である。また、スパース率が高い場合は、MF の使用は避けた方が良い。MF はスパース率が高くなると、推薦するアイテムがある特定のものに偏ってしまうからである。最後に、ユーザごとに個別化された推薦ができることを重視するのであれば、iCTR を用いるほうが良い。iCTR は、推薦数やスパース率が変わっても、安定して高い IUD を保つからである。

6. おわりに

本研究では、協調フィルタリングとトピックモデルを組み合わせた代表的モデルである Collaborative Topic Regression (CTR) モデルに注目し、アイテムに関するテキストだけでなくユーザに関するテキストを利用することを考え、これらとベースラインである Matrix Factorization (MF) との比較を行う実験を行った。実験では、評価値の欠損にどれだけ対応できるかに重点を置き、評価値行列のスパース率を変化させた際の iCTR, uCTR と MF の性能を調べた。それぞれの手法を総合的な観点から評価を行うために、正確性に関する評価指標として再現率を、利便性に関する評価指標として被覆率と多様性を用いた。

実験の結果、どちらの CTR モデルも MF に比べて評価値の欠損による影響が小さくロバストであることが示せた。

iCTR と uCTR のどちらが優れているかに関しては一概に決めることは難しく、iCTR では個別化の高い推薦、uCTR では正確性の高い推薦を行う傾向が見られた。また、MF はスパース率が低い場合は最も正確性の高い推薦が行えたが、スパース率に対してきわめて敏感であるため、利用したいデータセットのスパース率が高いことが見込まれる場合には、iCTR や uCTR の導入を検討したほうが良いことが分かった。

今後の課題としては、ハイパーパラメータ間の交互作用、すなわちトピック数 K とモデルの複雑さを調整する λ_u , λ_v の設定を変化させた場合に、非常にスパースなデータセットとそうでないデータセット間に違いが生じるか調査する必要がある。また、CiteULike 以外のサービスにおけるデータを用いた実験も行い、発見した傾向の一般性を確かめる必要がある。

謝辞 本研究は科研費 (15K12150) の助成を受けたものである。

参考文献

- [1] 土方嘉徳：嗜好抽出と情報推薦技術，情報処理学会誌，Vol.48, No.9, pp.957–965 (2007).
- [2] Koren, Y., Bell, R. and Volinsky, C.: Matrix Factorization Techniques for Recommender Systems, *Computer*, Vol.42, No.8, pp.30–37 (2009).
- [3] Koren, Y.: Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model, *Proc. KDD*, pp.426–434 (2008).
- [4] Paterek, A.: Improving Regularized Singular Value Decomposition for Collaborative Filtering, *Proc. ACM SIGKDD Cup and mWorkshop* (2007).
- [5] Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and Metrics for Cold-Start Recommendations, *Proc. ACM SIGIR*, pp.253–260 (2002).
- [6] Chang, S., Harper, M.F. and Terveen, L.: Using Groups of Items for Preference Elicitation in Recommender Systems, *Proc. CSCW*, pp.1258–1269 (2015).
- [7] Wang, C. and Blei, D.M.: Collaborative Topic Modeling for Recommending Scientific Articles, *Proc. ACM SIGKDD*, pp.448–456 (2011).
- [8] Purushotham, S., Liu, Y. and Kuo, C.-C.J.: Collaborative Topic Regression with Social Matrix Factorization for Recommendation Systems, *Proc. ICML*, pp.759–766 (2012).
- [9] Ding, X., Jin, X., Li, Y. and Li, L.: Celebrity Recommendation with Collaborative Social Topic Regression, *Proc. IJCAI*, pp.2612–2618 (2013).
- [10] Kim, Y. and Shim, K.: TWILITE: A Recommendation System for Twitter using a Probabilistic Model based on Latent Dirichlet Allocation, *Journal of Information Systems*, Vol.42, pp.59–77 (2014).
- [11] 西村章宏，土方嘉徳，三輪祥太郎，西田正吾：一般ユーザの観点に基づく Twitter からの人物関係の可視化と事例の考察，情報処理学会論文誌，Vol.56, No.3, pp.972–982 (2015).
- [12] Su, X. and Khoshgoftaar, T.M.: A Survey of Collaborative Filtering Techniques, *Journal of Advances in Artificial Intelligence*, Vol.2009, No.4 (2009).
- [13] Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P. and Riedl, J.: GroupLens: An Open Architecture for Collaborative Filtering of Netnews, *Proc. ACM CSCW*, pp.175–186 (1994).
- [14] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J.: Item-based Collaborative Filtering Recommendation Algorithms, *Proc. ACM WWW*, pp.285–295 (2001).
- [15] Canny, J.: Collaborative Filtering with Privacy via Factor Analysis, *Proc. ACM SIGIR*, pp.238–245 (2002).
- [16] Hofmann, T. and Puzicha, J.: Latent Class Models for Collaborative Filtering, *Proc. IJCAI*, pp.688–693 (1999).
- [17] Cremonesi, P., Koren, Y. and Turrin, R.: Performance of Recommender Algorithms on Top-N Recommendation Tasks, *Proc. ACM RecSys*, pp.39–46 (2010).
- [18] Blei, D.M., Ng, A.Y. and Jordan, M.I.: Latent Dirichlet Allocation, *The Journal of Machine Learning Research*, Vol.3, pp.993–1022 (2003).
- [19] Burke, R.: Hybrid Recommender Systems: Survey and Experiments, *User Modeling and User-Adapted Interaction*, Vol.12, No.4, pp.331–370 (2002).
- [20] Popescul, A., Ungar, L.H., Pennock, D.M. and Lawrence, S.: Probabilistic Models for Unified Collaborative and Content-based Recommendation in Sparse-data Environments, *Proc. UAI*, pp.437–444 (2001).
- [21] Kim, B.M. and Li, Q.: Probabilistic Model Estimation for Collaborative Filtering based on Items Attributes, *Proc. IEEE/WIC/ACM WI*, pp.185–191 (2004).
- [22] Salakhutdinov, R. and Mnih, A.: Probabilistic Matrix Factorization, *Proc. NIPS 21* (2008).
- [23] Hu, Y., Koren, Y. and Volinsky, C.: Collaborative Filtering for Implicit Feedback Datasets, *Proc. IEEE ICDM*, pp.263–272 (2008).
- [24] Bertsekas, D.P.: *Nonlinear Programming, 2nd Edition*, Athena Scientific, Belmont, Massachusetts (1999).
- [25] 土方嘉徳：推薦システムのオフライン評価手法，人工知能学会誌，Vol.29, No.6, pp.658–689 (2014).
- [26] Ge, M., Delgado-Battenfeld, C. and Jannach, D.: Beyond Accuracy: Evaluating Recommender Systems by Coverage and Serendipity, *Proc. ACM RecSys*, pp.257–260 (2010).
- [27] Herlocker, J.L., Konstan, J.A., Terveen, L.G. and Riedl, J.T.: Evaluating Collaborative Filtering Recommender Systems, *ACM Trans. Information Systems (TOIS)*, Vol.22, No.1, pp.5–53 (2004).
- [28] Adomavicius, G. and Kwon, Y.: Improving Aggregate Recommendation Diversity Using Ranking-Based Techniques, *IEEE Trans. Knowledge and Data Engineering*, Vol.24, No.5, pp.896–911 (2012).
- [29] Zhou, T., Kuscsik, Z., Liu, J.-G., Medo, M., Wakeling, J.R. and Zhang, Y.-C.: Solving the Apparent Diversity-accuracy Dilemma of Recommender Systems, *Proc. National Academy of Sciences*, Vol.107, No.10, pp.4511–4515 (2010).
- [30] Lathia, N., Hailes, S., Capra, L. and Amatriain, X.: Temporal Diversity in Recommender Systems, *Proc. ACM SIGIR*, pp.210–217 (2010).
- [31] Griffiths, T.L. and Steyvers, M.: Finding Scientific Topics, *Proc. National Academy of Sciences*, Vol.101, pp.5228–5235 (2004).



西村 章宏

2013年大阪大学基礎工学部システム科学科卒業。2015年同大学大学院修士課程修了。第159回情報処理学会データベースシステム研究会・学生奨励賞受賞。在学中は、情報可視化と情報推薦の研究に従事。



土方 嘉徳 (正会員)

1996年大阪大学基礎工学部システム工学科卒業。1998年同大学大学院修士課程修了。同年日本アイ・ビー・エム(株)東京基礎研究所入社。2002年より大阪大学大学院基礎工学研究科システム創成専攻助手。2009年より同准教授。2014年ミネソタ大学 GroupLens Research 客員研究員。2017年より関西学院大学商学部准教授。現在に至る。2005年インタラクシオン 2005 ベストペーパー賞, 2006年 ACM IUI Best Paper Award, DEWS2006 優秀論文賞, 2011年 WebDB フォーラム 2011 最優秀論文賞, 2012年 WebDB フォーラム 2012 優秀論文賞, 2013年インタラクシオン 2013 ベストペーパー賞, WebDB フォーラム 2013 優秀論文賞, 情報処理学会山下記念研究賞各受賞。ソーシャルコンピューティング, 情報推薦, データマイニングの研究に従事。人工知能学会, ヒューマンインタフェース学会, 日本データベース学会ほか各会員。電子情報通信学会シニア会員。博士(工学)。



酒田 信親

2002年筑波大学工学システム学類卒業。2007年同大学大学院システム情報工学研究科知能機能システム専攻博士課程修了。2012~2013年 HITLAB NZ University of Canterbury 客員研究員。2007年4月より大阪大学大学院基礎工学研究科システム創成専攻助教。現在に至る。実世界指向インタフェースの研究に従事。博士(工学)。著書は『次世代ヒューマンインタフェース開発の最前線』(NTS, 共著), 『AR(拡張現実)技術の基礎・発展・実践』(科学情報出版)。