情報処理学会論文誌 Vol.62 No.12 2108–2118 (Dec. 2021)

# Cold-Start問題と多様性を考慮したレコメンデーションの総合評価

内田 匠[1,a)] 吉田 健一[2,b)]

受付日 2021年2月26日, 採録日 2021年9月9日

**概要**：レコメンデーションの基本的な役割はユーザが満足する情報を推薦することであるが，それを評価する指標としては，推薦精度，Cold-Start 問題への対処，推薦内容の多様性など，様々な観点がある．従来の研究ではこれらの評価指標の 1 つに注目することが多く，総合的な評価研究は課題が残されていた．本研究ではレコメンデーションの総合的な評価方法を提案し，既存のレコメンデーション手法を評価することで，提案する評価方法の有用性を示す．提案した評価方法に基づき，既存のレコメンデーション手法を評価し，「従来，推薦困難とされていた新規ユーザへの推薦精度は比較的簡単に達成できる．一方で，利用頻度が多いユーザへの推薦精度と多様性はレコメンド手法で解決すべき課題となっている」ことを明らかにした．提案する評価方法の特徴は下記の 2 つである．(1) ユーザとアイテムを履歴データでの頻度から [new（新規），tail（低頻度），head（高頻度）] に分け，未来に対する推薦精度を比較することで推薦精度と Cold-Start 問題に対する性能を評価する．(2) 同時に，表示機会をアイテムごとに分配する度合い（多様性）も評価する．

キーワード：推薦システム，精度評価，多様性

# An Evaluation of Recommendations Considering the Cold-Start Problem and Diversity of Information Recommended

Takumi Uchida[1,a)] Kenichi Yoshida[2,b)]

Received: February 26, 2021, Accepted: September 9, 2021

***Abstract:*** The role of recommendation has two aspects: 1) accurate provision of information requested by users, and 2) diversity of information provided. Research on conventional recommendation methods often focus on the accuracy of the information provided, and research on diversity remains a problem. In this study, we propose an evaluation method of recommendation method that considers both accuracy and diversity. We show the importance of the proposed evaluation method by evaluating the existing recommendation methods' performance on the Cold-Start problem. Specifically, "With the existing recommendation method, it is relatively easy to achieve the accuracy of recommending the Cold-Start problem to new users. However, it is not easy to achieve the diversity of information provided to frequently used users. The features of the proposed evaluation method are the following: (1) Divide users and items into [new (new), tail (low frequency), head (high frequency)] according to the frequency in the historical data. Compare the recommendation accuracy for the future. (2) Compare the diversity of displayed items.

***Keywords:*** recommendation, off-line verification

## 1. はじめに

レコメンデーションの役割には，ユーザが求める情報の推薦だけではなく，Cold-Start 問題への対処，推薦する情報の多様性などがある．従来のレコメンデーション手法の研究は推薦精度を中心に評価されることが多く，多様性な

[1] インキュデータ株式会社
INCUDATA Corp., Chuo, Tokyo 104–0061, Japan
[2] 筑波大学大学院ビジネス科学研究科
Graduate School of Business Science, University of Tsukuba, Bunkyo, Tokyo 112–0012, Japan
[a)] takumi.uchida.research@gmail.com
[b)] yoshida.kenichi.ka@u.tsukuba.ac.jp

どの複数の観点から評価する研究には課題が残されていた．たとえば，推薦精度だけでレコメンデーションを評価すると，ユーザがすでに知っている過去の人気アイテムに表示が偏ってしまう．ユーザにとって既知の人気アイテムばかりが推薦されても新たなアイテムを発見できないため，提供される情報の多様性は重要である．また多様性には，ユーザから見た推薦アイテムの多様性とアイテム提供者から見たアイテムの表示機会の分配を目的とした多様性がある．前者の多様性はユーザの満足度に影響し，後者は在庫や新商品の育成などのアイテム提供者側の目的として重要な指標である．このような多様性の課題は “Beyond Accuracy” [1] と呼ばれており，重要な研究テーマとなっている．

また，Cold-Start 問題への対処もレコメンデーションの課題である．Cold-Start 問題とは「過去の利用履歴データが少ないユーザと情報の推薦が困難な問題」と一般に定義されている．適切な推薦が行われない場合，ユーザはアプリケーションの利用をやめ，新しいアイテムに関する情報は誰にも推薦されることなく埋もれてしまう危険性がある．よって Cold-Start 問題に対する性能も重要な評価指標である．

本研究では既存のレコメンド手法に対して，Cold-Start 問題に対する性能をユーザとアイテムの両軸で比較し，それぞれの推薦多様性も検証する．検証する際には，既存研究が公開した実装や設定に極力準拠した．また他の Cold-Start 問題でも用いられている MovieLens の 20M データセットを用いて，提案する検証法で既存のレコメンド手法を比較し，Cold-Start 問題の新規の解釈を発見した．

以降では，2 章で関連研究を概観した後，3 章で提案する評価方法を説明し，4 章でその結果を報告する．

## 2. 関連研究

### 2.1 レコメンデーションの評価方法

レコメンデーションの評価方法はオンライン評価とオフライン評価に大きく区別される．オンライン評価は実際に利用されているアプリケーション上で行われるもので，A/B テストなどが代表的である．対して，オフライン評価は過去のデータに対して行われる．一般にオンライン評価のほうがより実践的であるが，その実施コストは高く試行回数にも限界がある．それを補うためにオフライン評価は重要である．本研究はオフライン評価の方法を提案するため，以降ではオフライン評価の先行研究をまとめる．

オフライン評価法では，指標と手順の組合せでその性質が決まる．指標として，平均絶対値誤差（MAE），二乗平均平方根誤差（RMSE），適切なアイテムが推薦リストに含まれているかを評価する ROC（Receiver Operating Characteristic）など推薦精度の評価指標が扱われることが多い．これらの精度指標とは別に推薦アイテムの多様性を評価する指標も提案されている．手順とは，訓練とテストセットを分離する手順である．多くの研究ではランダムでデータセットを分離している．この際，Cold-Start 問題への性能を検証する際には，意図的に訓練セットのデータ量を少なくサンプリングするなどの工夫を行う．

この評価に用いる指標と手順によって手法に対する評価結果が変わることが報告されている．たとえば，文献 [1] は，指標を RMSE とした場合と ROC とした場合を比較し，その評価結果にまったく相関性がなかったことを報告している．また，文献 [2] は後述するより厳密な手順を提案し，そのうえで既存のレコメンド手法を再評価した結果，当時の最新手法であった SVD++ [3] が従来の MF 法よりも劣ったことを報告している．

上述のように，オフライン評価では評価目的に応じた指標と手順の選択が重要となる．本研究では推薦精度だけでなく Cold-Start 問題と多様性を考慮したレコメンデーション手法のオフライン評価方法を提案する．後述する既存研究と本研究で提案する評価法の違いを**表 1** に示す．

#### 2.1.1 推薦精度指標と手順

代表的な精度指標は ROC 曲線などを用いて，ユーザが満足した（たとえば，ユーザが最高スコアの 5 点をつけた）アイテムが top-N 推薦リストに含まれているかを評価する．文献 [2] はその評価手順を以下のように定義した．

(1) ユーザ $u$ が高いスコアをつけたアイテム集合 $I$ を抽出し，そのアイテム $i \in I$ を $u$ への top-N 推薦リストに含めるべき正解とする．
(2) ユーザ $u$ が未選択のアイテムをランダムで 1,000 個抽出する．このランダム抽出されたアイテムは不正解と見なす．
(3) 正解のアイテム $i$ と不正解である 1,000 個のアイテムについて，ユーザ $u$ のレビュースコアを評価したいレコメンド手法で予測する．
(4) 1,001 個のアイテムを予測値が大きい順に並べる．このとき，正解であるアイテム $i$ の順番を $p$ とする．最も良い結果の場合は $p = 1$ となり，$p$ が大きくなるほど推薦精度は低いといえる．
(5) $p <= N$ であれば表示リスト内に正解アイテムが存在するため，テスト $(u, i)$ の結果を $hit = 1$ とする．$p > N$ の場合は $hit = 0$ とする．
(6) 上記の処理を，すべてのテストセット $T$ について繰り返す．

文献 [2] は上記の手順の後に，以下の Recall，Precision を精度指標としている．

$$Recall(N) = \frac{\sum_T hit}{|T|} \tag{1}$$

$$Precision(N) = \frac{\sum_T hit}{N \cdot |T|} = \frac{recall(N)}{N} \tag{2}$$

表 1 オフライン評価法の既存研究と本研究の立ち位置
**Table 1** Related studies of offline evaluation methods and the position of this study.

| 論文 | 手順 | 指標 | 補足 |
|---|---|---|---|
| 文献 [1] | ランダムで訓練/テストセットを分ける. | 精度指標（MAE, ROC, Pearson） | レコメンデーションの評価方法のサーベイ研究．複数のレコメンド手法を様々な指標で再評価し，指標によってその優劣が変化することを確認. |
| 文献 [2] | ランダムで訓練/テストセットを分ける.<br>最大スコアを獲得したアイテムを正解とし，ランダム選出した不正解アイテムの中から，N-top リスト内に含めることができるかを確認. | 精度指標（ROC） | ROC をより厳密に計算するための手順を提案し，既存の手法を再評価. |
| 文献 [4] | ランダムで訓練/テストセットを分ける. | 多様性指標（Catalog Coverage） | 文献 [1] で提案されている多様性指標を定義し，その結果を報告. |
| 文献 [5] | ランダムで訓練/テストセットを分ける. | 多様性指標（Intra-List Similarity） | 1 つの推薦リスト内に含まれるアイテム群が，十分に多様であるかを評価する指標を提案. |
| 文献 [6] | ランダムでユーザ ID を訓練とテストに分け，訓練を既存ユーザ，テストを新規ユーザと見なす. | 精度指標（RMSE） | 新規ユーザの Cold-Start 問題に特化した評価法を提案. |
| 文献 [7] | 過去と未来で訓練/テストセットを分ける. | 精度指標（RMSE） | レコメンドの学習による性能更新を意識しており，学習セットは過去のすべてを対象とし検証後半になるにつれデータ量は増加している. |
| 本研究 | 過去と未来で訓練/テストセットを分ける.<br>訓練セット内の出現頻度に応じて，ユーザとアイテムを [head，tail，new] と定義.<br>最大スコアを獲得したアイテムを正解とし，ランダム選出した不正解アイテムの中から，N-top リスト内に含めることができるかを確認. | 精度指標（Recall）と<br>多様性指標（Entropy） | |

#### 2.1.2 多様性指標

文献 [1] は，レコメンデーションの多様性指標として Catalog Coverage を提案した．この具体的な計算を文献 [4] は以下のように定義した.

$$CatalogCoverage = \frac{|\cup_{j=1\ldots J} L_j|}{|I|} \tag{3}$$

$I$ はアイテムの全集合，$L_j$ はテストで出力した $j$ 回目の推薦リストに含まれる item 集合，$J$ はレコメンデーションが行われた総数である．よって，Catalog Coverage は計測期間中に 1 度でも表示された item の和集合と item の全集合の比率である.

一方で，ユーザ視点での多様性指標として，文献 [5] で提案されている Intra-List Similarity がある．これは推薦リスト内のアイテムの類似度が低いほど多様性が高いと定義している.

### 2.2 Cold-Start 問題

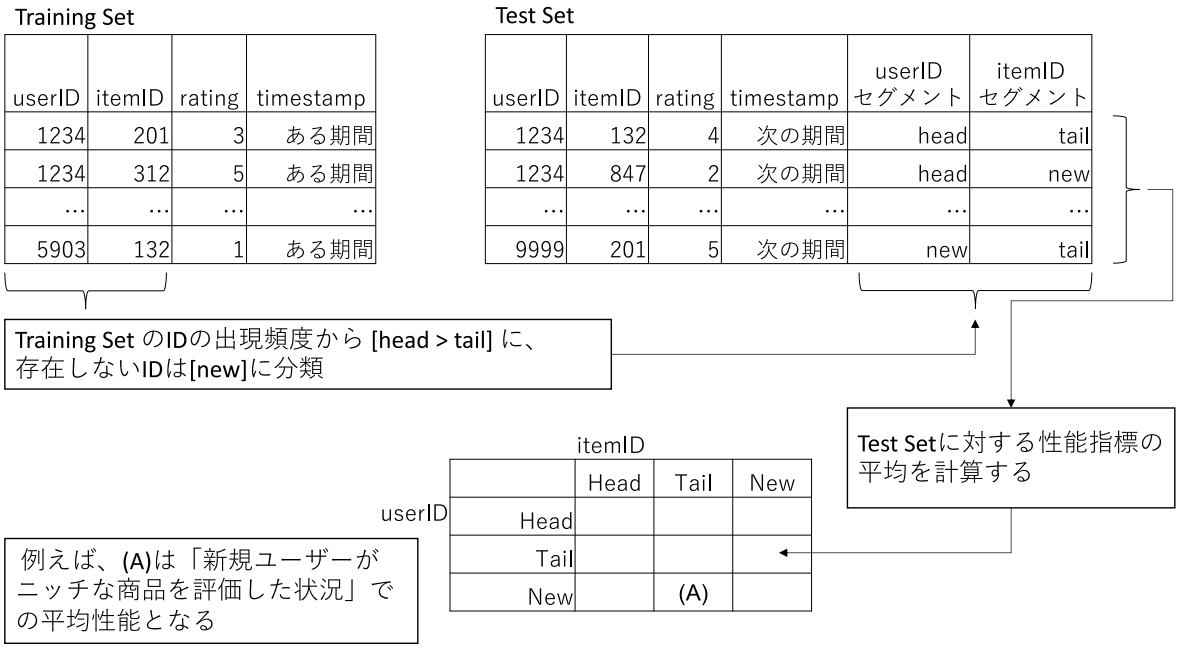
レコメンデーションの研究では，MovieLens などのデータを User-Item Matrix（**表 2**）に変換して計算をすることが多い．User-Item Matrix とは X 軸にユーザ ID，Y 軸にアイテム ID を並べた行列であり，その要素は満足度スコアや購買の有無である．一般的に，ユーザはすべてのアイテム ID を体験することはなく，この行列の要素はほとんどが未観測の欠損値となっている．レビュー数の少ないユーザではアイテムの欠損値が多くなるため予測が困難であり，さらにまったく新規の ID に関する予測は不可能となる.

表 2 User-Item Matrix の例
**Table 2** An example of User-Item Matrix.

| | **item0** | **item1** | **item2** | **item3** | ··· |
|---|---|---|---|---|---|
| **user0** | 3.0 | null | null | null | ··· |
| **user1** | null | null | null | null | ··· |
| **user2** | null | 5.0 | null | 2.5 | ··· |
| **user3** | null | null | null | null | ··· |
| ··· | ··· | ··· | ··· | ··· | ··· |

観測されたユーザとアイテムのレビュースコアを行列で表現したもの．null は未観測の欠損値であり，レビューがないことを表している.

この欠損値が Cold-Start 問題の主な原因であり，それを解決するために User-Item Matrix だけではなく，アイテムやユーザの属性情報などの Addtional Data も学習することが提案されている．たとえば，映画のドキュメント情報を tf-idf [8] でベクトル化して Addtional Data とする手法 [9] などがある．この他にも，多くの研究が Additional




ID の分類数を $(Y+1)=3$ の設定にした場合を図にした．訓練セットに多く出現する ID は head，少ないのは tail，テストセットで新しく出現した ID を new と命名している．

**図 1** 表示精度評価の概要
**Fig. 1** Overview of display accuracy evaluation.

**表 3** レコメンド手法の分類表
**Table 3** Classification of recommender systems.

| | only User-Item Matrix | with Additional Data |
|---|---|---|
| Memory-Based | Item-Based [14]<br>User-Based [15] | Content-Boosted CF [16] |
| Model-Based | MF [17] | TWA [11]<br>MF |
| 深層学習応用 | DFM [18] | |

縦軸が計算手法による分類で，横軸が利用するデータによる分類．Additional Data の具体例は，年齢や職業などのユーザ属性や，映画のあらすじテキストやジャンルなどのアイテム属性など．

Data を用いた手法を提案し Cold-Start 問題の改善を報告している（文献 [10], [11], [12] など）．

### 2.3 レコメンド手法の分類

Larson らは伝統的なレコメンド手法の分類である Memory-Based，Model-Based の 2 分類に，データが User-Item Matrix のみかと Addtional Data を追加するかの分類を加え，4 分類とした [13]．また，近年では深層学習をレコメンデーションに応用する研究が進んでいる．

以下の章では，伝統的な分類である Memory-Based，Model-Based に深層学習 Based を加えた 3 分類に，Additional Data の有無を考慮した 6 分類（**表 3**）の手法を提案法により評価する．

## 3. 提案するオフライン評価法

### 3.1 評価手順と精度指標

既存研究の性能評価で用いられるクロスバリデーションでは時間順を無視して訓練/テストセットをランダムに分割するため，未来のデータを含んだデータで訓練したレコメンド手法を過去のデータで性能計測することになる点で不適切である．そこで提案する評価方法では，文献 [7] の研究と同様に，過去 $X$ 日間のデータを用いてレコメンド手法を訓練し，直後の $X$ 日間での性能を計測する．

より詳細には**図 1** に示した以下の手順に従う．

(1) データを特定期間 $X$ ごとに分割し，ある期間を訓練セット，次の期間をテストセットとする．
(2) 訓練セットでの ID の出現頻度に応じて，テストセットのユーザ ID とアイテム ID を $Y+1$ 個に分類する．+1 は訓練セットに存在しない ID の分類で new（新規）とする．$Y=2$ であれば ID を出現頻度の多い順に head（高頻度），tail（低頻度）の 2 つと new の 3 つに分類する[*1]．
(3) 訓練セットを使い評価対象のレコメンド手法を訓練する．
(4) 1 つ 1 つのテストサンプル $(u,i)$ から閾値 $Z$ 以上のレビュースコアのものだけで，Recall（式 (1)）もしくは Precision（式 (2)）を精度指標として計算する．

[*1] 提案手法は 3 以上の $Y$ でも動作するが，以下実務上特に重要な $Y=2$，すなわち head，tail，new の 3 分類について説明する．

ここでテストサンプル $(u,i)$ は，ユーザ $u$ がアイテム $i$ につけたレビュースコアを表す．

(5) $Y=2$ であれば，テストサンプル $(u,i)$ は [head×head, head×tail, $\cdots$] のように $(Y+1)^2=9$ 個に分類できる．これを評価セグメントとし精度の平均値を算出する．

(6) 上記の計算手順を $X$ だけ期間をずらしながら繰り返し（繰り返しごとの学習データには重複する期間はない），その平均値をその評価対象手法の表示精度とする．

期間 $X$，ID の分類数 $Y+1$，閾値スコア $Z$ は事前に設定すべきパラメータである．これらのパラメータは，分析するデータ量やアプリケーションの性質によって個別に設定する．たとえば，ファッション通販などでは季節性が強いため $X$ を 60 日と長く設定すると訓練セットとテストセットで季節が変わってしまい，適切に手法を評価できない．また，一般に機械学習では訓練データが多いほど性能が高くなることが期待されるが，レコメンデーションではそうとは限らない．直近過去の短い期間で訓練した場合，推薦時点の流行を学習しやすくなり，もう見られなくなった過去の人気アイテムに過大な評価をしなくなるなどの短い学習期間だからこその利点もある．上述のような点に注意し，それぞれのアプリケーションに応じて $X$，$Y$，$Z$ を適切に設定する必要がある．

提案手法は上記の評価手順によって「head ユーザに new アイテムを推薦する性能」など [head，tail，new] のユーザ × アイテムの合計 9 セグメントでの精度を評価する．

### 3.2 表示機会の分配を評価する多様性指標

本研究では多様性評価の指標としてエントロピーを用いることを提案する．たとえば，100 人のユーザに対して Top10 リストを表示した場合はのべ 1,000 回の表示機会がある．各アイテム $i$ が $p_i$%の確率で表示機会を得た場合のエントロピー $E$ は以下のように計算する．

$$E=-\sum_{i\in I} p_i \log p_i \tag{4}$$

$I$ はアイテムの全集合であり，多くのアイテム $i$ が均等に表示割合 $p_i$ を得ているほど $E$ は大きくなる．

先行研究の Catalog Coverage（式 (3)）が 1 度でも表示されたアイテムの種類数を評価しているのに対して，エントロピーは表示回数の均等な分配も考慮した指標となる．文献 [1], [5] の Catalog Coverage と同様に，エントロピーはアイテム提供者視点での多様性指標であり，1 人のユーザが多様なアイテムの推薦を受けたかの指標にはならない点には注意が必要である．

## 4. 評価実験の結果

提案する評価方法を用いて 2.3 節で取り上げた代表的なレコメンド手法を比較する．比較したレコメンド手法のうち，TWA 法と Content-Boosted CF 法は User-Item Matrix 以外の属性データも学習することで，Cold-Start 問題に対処する手法である．また，DFM 法にもアイテム属性を学習させ，同様の考えで Cold-Start 問題に対処させた．

### 4.1 評価に用いたデータ

既存の Cold-Start 問題の研究にも利用されている（文献 [19], [20], [21]）MovieLens の映画スコアデータ [22] のうち，研究目的利用に奨励されている 20M データセット[*2]を用いた．このデータはユーザが映画に 0.5 から 5.0 の 0.5 刻みのレビュースコアをつけたデータで，集計期間は 1995/01/09 から 2015/03/31 まで，27,278 の映画に対して 138,493 人のユーザによる 20,000,263 のレビューがある．

また，このデータセットは 20 以上の映画にレビューしたユーザのみを抽出している．これは一般的な映画に関するアプリケーションではヘビーユーザに該当するが，本評価では訓練/テストセットを前後 30 日間ごとに分け，区分内のデータ量を減らすことで head，tail，new のユーザを再現した．new に分類されるユーザとアイテムは，過去 30 日でレビューがなかっただけでそれより過去にはレビューしたものも含めた．つまり，本評価における new は，訓練期間中の未学習 ID に対する評価セグメントである．

### 4.2 評価実験の設定

3.1 節の精度評価に必要な設定を，期間 $X$ を 30 日，ID の分類数 $Y+1=3$ として head，tail，new を評価した．ユーザが満足したアイテムを判断するレビュースコアの閾値 $Z$ は，文献 [2] に準拠し，確かにユーザが満足したアイテムのみで評価するため最高スコアである 5 を採用した．評価に用いるのは映画のデータであるため，新作の公開日を起点とする一定の流行性があると考えられる．映画の興業収入ランキングでは，1 週間ごとに上位は入れ替わり，1 つの映画が 1 カ月以上も上位に入ることは少ない．このことから，本評価では $X$ を 30 日と設定した．

ID の分類数 $Y$ は多いほど詳細な評価が可能になるが，多すぎると評価に十分なデータ量が確保できない．本実験では $Y+1=3$ 分類とした．さらに，結果を安定させるため，評価の繰り返し数を 60 回とし，$30\times60$ 日間の約 6 年間のデータで評価を行った．

また head/tail の分離点によって結果が変化するか確認するため，出現頻度の上位 10，15，20%とした場合をそれぞれ確認したが，それぞれの結果に大きな違いはなかったので，以下 15%の結果のみを報告する．また，この設定における tail ユーザの User-Item Matrix の観測された要素の割合は 0.146%であり，tail アイテムの場合は 0.045%で

[*2] https://grouplens.org/datasets/movielens/20m/

ある．それぞれ 99.8%以上が欠損値であり Cold-Start 問題が懸念される推薦状況となる．

### 4.3 対照手法

レコメンド手法に対する対照手法として，人気ランキング表示を検証に加える．人気ランキング表示は，レコメンド手法と違ってすべてのユーザに同じアイテムを表示し，計算が比較的簡単であり，多くのアプリケーションで用いられている．そのため，対照手法として適切であると判断した．検証に用いた人気ランキング表示は訓練期間でスコアに関係なくレビュー数が多いアイテムを降順に並べる．本実験では訓練期間を 30 日に設定しているので，これは前月のレビュー数ランキングに相当する．

### 4.4 各レコメンド手法のチューニング

ここでは 2.3 節でとりあげたレコメンデーション手法にどのような調整をしたのかを述べる．

手法によっては Additional Data を必要とするが，本実験では MovieLens が公式に提供している映画のジャンル情報のみを用いた．これは Action や Comedy などの 18 種類のジャンルに属することを 0，1 で表したベクトルであり，1 つの映画が複数のジャンルに属することもある．用いる Additional Data によってレコメンド手法の精度は変わるが，本実験ではあくまでも MovieLens が公開しているジャンル情報の範囲で傾向を確認した．

また Additional Data として，ユーザの属性情報は用いないことにした．現在，個人情報保護の社会要請が高まっており，なるべくユーザ情報を用いないレコメンデーションが求められている．MovieLens でも 2016 年 10 月のアップデート以降，ユーザの年齢，性別，職業といった属性情報の提供を停止している．

各手法のハイパーパラメータの設定，実装方法，計算量オーダ，必要メモリ量を**表 4** に示す．これらのパラメータ設定，評価データに用いた全期間よりもさらに過去の 30 日間のデータで，グリッドサーチやランダムサーチを用いて探索した結果である．以降では，表 4 に記載していないより詳細な設定について述べる．

**Item-Based，User-Based 法**

本手法ではユーザ，アイテムのいずれかが new の場合は予測できないため，訓練データ全体の平均スコアを予測値とした．

**Content-Boosted CF 法**

本実験では Additional Data として映画のジャンル情報のみを用いるため，未観測のレビュースコアの推定には文献 [16] が用いた式ではなく，下記の式 (5) を用いた．

$$\hat{R}_{ui} = \mu + b_u + b_i + \frac{wA_{i*}}{\sum_{a \in A_{i*}} A_{ia}} \tag{5}$$

ここで，$\mu$ はグローバルバイアス，$b_u$ はユーザ $u$ のバイアス，$b_i$ はアイテム $i$ のバイアス，$A_{i*}$ はアイテム $i$ のジャンル情報ベクトル，$w$ はその回帰係数ベクトルである．

本手法はアイテムが new の場合は予測可能だが，ユーザが new の場合は不可能である．そこで new ユーザの予測確率（TWA の予測値は確率になる）は平均値を用いた．

**MF 法，MF（with genres）法**

ユーザ，アイテムのいずれかが new の場合，MF が仮定する潜在変数ベクトルとバイアスの値を 0 として計算した．

**TWA 法**

new アイテムに対して予測が可能であるが，ユーザが new の場合は不可能である．その場合は平均確率を予測値とした（TWA の予測値はスコアではなく確率である）．

**DFM 法**

Deep Factorization Machine（DFM）法を提案した文献 [18] はレーティングスコアではなく，$[0, 1]$ で計測されるクリックを学習している．そのため，MovieLens のレビュースコアデータを二値に変換する必要がある．この問題に対して文献 [25] や [26] では，観測された $(u, i)$ を正例とし，ユーザが未体験のアイテムをランダム抽出し負例サンプリングを実施している．この負例サンプリングは，ユーザが未体験のアイテムのほとんどをユーザは求めていないという仮説にもとづいている．この負例サンプリングには以下の 2 種類が考えられる．

- レビュースコアが最大値の 5 だけを正例とし，それと同じ量の負例サンプルをランダム生成する．
- すべてのレビュースコアを正例とし，それと同じ量の負例サンプルをランダム生成する．

前者の負例サンプリングが正例のデータの信頼性，後者はデータ量を重視する学習である．実際に検証した結果，精度評価が最も高かった後者を後述の実験結果に記載した．

### 4.5 精度評価の結果

head，tail，new のアイテム ID ごとに，精度指標として Top10 レコメンドリストの recall を**表 5** に示す．

表の recall は 1,000 個のアイテム中から 1 つの正解を Top10 リストに含める確率である．仮にランダムに推薦したとすると，recall は 0.01 になるため，それよりも十分に大きい場合に有効な推薦がされたといえる．

ランキング表示は，全ユーザ，head，tail，new の各ユーザに対しても最も平均精度が高かった（表 5）．また，ランキング表示では，ユーザが head，tail，new と過去データが少ないほど平均精度が改善していた．一般的に Cold-Start 問題では，過去データが少ないほど推薦精度が悪化するとされているが，ランキング表示では逆に良くなっている．同様の傾向は，MF 法と DFM 法でも確認できる．一方で，アイテムについては head，tail，new になるほど精度がおおむね悪化する傾向が確認でき，通常の Cold-Start 問題の

表 4 評価に用いた表示手法の設定や実装など
**Table 4** Configuration and implementation of the display methods used in the evaluation.

| 表示手法 | 設定/ハイパーパラメータ | 実装方法 | 訓練時の計算量オーダ | 必要メモリ量 |
|---|---|---|---|---|
| ランキング | 過去 30 日間でレビュー数が多いアイテムを降順に出力. | 独自に実装 | $O(\|I\|)$ | $\|I\|$ |
| Item-Based | 近傍数は 50. 類似度関数はコサイン類似度関数. | 文献 [23] の実装を利用 | $O(\|I\| \times \|I\|)$ | $\|I\| \times \|I\|$ |
| User-Based | 近傍数は 50. 類似度関数はコサイン類似度関数. | 文献 [23] の実装を利用 | $O(\|U\| \times \|U\|)$ | $\|I\| \times \|I\|$ |
| Content-Boosted CF | 欠損値の補完に用いる Content-Based は式 (5) を採用. 補完された結果を学習する User-Based の近傍数は 50. | 独自に実装 | $O(\|I\| \times \|I\|) + O(\|U\| \times \|I\|) + O(\|U\| \times \|U\|)$<br>第 1 項はコンテンツベースモデルの学習<br>第 2 項は User-Item Matrxi の欠損値補完<br>第 3 項は補完結果を User-Based で学習 | $\|U\| \times \|U\|$ |
| MF | 潜在因子数は 200，epoch 数は 20，学習率は 0.005，L2 正則化係数は 0.02 | 文献 [23] の実装を利用 | $O(epoch \times \|R\|)$ | $factor \times (\|U\| + \|I\|)$ |
| MF（with genres） | 潜在因子数は 200，epoch 数は 20，学習率は 0.005，L2 正則化係数は 0.02 | 独自に実装 | $O(epoch \times \|R\|)$ | $factor \times (\|U\| + \|I\|)$ |
| TWA | 潜在クラス数は 50. 各 EM アルゴリズムの計算ステップ数は 200. | 独自に実装 | $O(step \times \|U\| \times \|A\| \times \|Z\|) + O(step \times \|I\| \times \|A\| \times \|Z\|)$<br>第 1 項は 1 段階目の EM アルゴリズム<br>第 2 項は 2 段階目の EM アルゴリズム | $\|Z\| \times (\|U\| \times \|A\| + \|U\| + \|A\|) + \|Z\| \times (\|I\| \times \|A\| + \|I\| + \|A\|)$ |
| DFM | 潜在因子数は 8，epcoh 数は 30，fm 層と deep 層を使用. deep 層は [32,32] の 2 層で 50%の dropout. 学習率は初期値 0.001 の Adam アルゴリムで調整. | 文献 [24] の実装を利用 | $O(epoch \times \|R\|)$ | $factor \times (\|U\| + \|I\|)$ |
| DFM（with genres） | 潜在因子数は 8，epcoh 数は 30，fm 層と deep 層を使用. deep 層は [32,32] の 2 層で 50%の dropout. 学習率は初期値 0.001 の Adam アルゴリムで調整. | 文献 [24] の実装を利用 | $O(epoch \times \|R\|)$ | $factor \times (\|U\| + \|I\| + \|A\|)$ |

$|U|$ はユーザ数，$|I|$ はアイテム数，$|R|$ は観測されたレビュー数，$|A|$ は用いる Additional Data の変数の数，$|Z|$ は TWA 法が用いる潜在クラスの数，$epoch$ と $step$ は繰り返し訓練を行う回数，$factor$ は潜在因子の数である．公開されている適切な実装がなかった場合は，その手法の提案論文の内容に従って独自に実装した．

傾向が確認できた．これは従来の Cold-Start 問題の定義とは異なる結果である．つまり，映画レビューのデータではユーザ側には Cold-Start 問題は発生せず，単純なランキング表示で高い精度が達成できている．この要因としては，映画アプリケーションではベストセラーであることがユーザの行動に影響を与えていることが考えられる．

深層学習を応用した DFM 法はレコメンデーション手法の中では，平均的に高い精度となっている．表 5 から DFM はランキングより少し低い精度であるが，tail，new アイテムの推薦精度が 0 であることは同じであることが分かる．また，後述するように DFM 法の多様性はランキングとほぼ同じ水準にある（**表 6**）．精度と多様性が似ていることから，DFM 法がランキングと似たような推薦をしていることが懸念される．その要因として，本評価では DFM 法を負例サンプリングで学習したこともあり，正例として多く学習される人気アイテムの影響を強く受けたことが考えられる．また，Additional Data を用いた場合の tail，new のアイテム ID での性能を期待していたが，その精度はほぼ 0%で機能していない．

MF 法も DFM 法と同様に平均的な精度が高く，エントロピーも比較的高いため表示機会を多様なアイテムに分配しているといえる．ただし，Additional Data を用いた場合でも tail，new のアイテムを推薦する精度は 0%である．

TWA 法は Additional Data を用い，new のアイテムの推薦で高い精度となった．head，tail のユーザに対する new のアイテムを推薦する精度は基準値である 0.01 よりも十

表 5 Top10 レコメンドリストの Recall
**Table 5** Recall of the Top 10 recommendation list.

| 【全ユーザ】 | | | アイテム種別 | | | |
|---|---|---|---|---|---|---|
| 手法カテゴリ | 利用するデータ | 手法名 | all | head | tail | new |
| 対照手法 | only UI-matrix | ランキング | **0.307** | **0.396** | 0.000 | 0.000 |
| Memory-Based | only UI-matrix | Item-Based | 0.010 | 0.010 | 0.010 | (0.008) |
| | | User-Based | 0.010 | 0.010 | 0.011 | (0.008) |
| | with Additional-data | Content-Boosted CF | 0.025 | 0.029 | 0.011 | 0.008 |
| Model-Based | only UI-matrix | MF | 0.236 | 0.303 | 0.012 | 0.000 |
| | with Additional-data | MF（with genres） | 0.230 | 0.294 | **0.014** | 0.000 |
| | | TWA | 0.012 | 0.012 | 0.011 | **0.013** |
| 深層学習応用 | only UI-matrix | DFM | 0.250 | 0.322 | 0.000 | (0.000) |
| | with Additional-data | DFM（with genres） | 0.243 | 0.313 | 0.000 | 0.000 |

表中の太字は最も精度が高い手法を表し，() つきの精度はその手法が予測できないセグメントで平均値などの代替値で予測していることを表す．不正解のアイテム ID 1,000 個から 1 つ正解を top10 リストに含める精度なので，ランダム表示の recall は 0.01 になる．

| 【head ユーザ】 | | | アイテム種別 | | | |
|---|---|---|---|---|---|---|
| 対照手法 | only UI-matrix | ランキング | **0.155** | **0.242** | 0.000 | 0.000 |
| Memory-Based | only UI-matrix | Item-Based | 0.003 | 0.000 | 0.009 | (0.000) |
| | | User-Based | 0.004 | 0.000 | 0.013 | (0.000) |
| | with Additional-data | Content-Boosted CF | 0.122 | 0.185 | 0.014 | 0.000 |
| Model-Based | only UI-matrix | MF | 0.090 | 0.136 | 0.015 | 0.000 |
| | with Additional-data | MF（with genres） | 0.088 | 0.132 | 0.016 | 0.000 |
| | | TWA | 0.024 | 0.022 | **0.027** | **0.026** |
| 深層学習応用 | only UI-matrix | DFM | 0.128 | 0.200 | 0.000 | (0.000) |
| | with Additional-data | DFM（with genres） | 0.130 | 0.203 | 0.000 | 0.000 |

head のユーザ ID の場合も全体的な精度はランキング表示が最も良い．tail/new アイテムの場合，ランキング表示は役にたたず，Memory-based/Model-base の手法が良い．

| 【tail ユーザ】 | | | アイテム種別 | | | |
|---|---|---|---|---|---|---|
| 対照手法 | only UI-matrix | ランキング | **0.199** | **0.316** | 0.000 | 0.000 |
| Memory-Based | only UI-matrix | Item-Based | 0.010 | 0.009 | 0.015 | (0.003) |
| | | User-Based | 0.014 | 0.012 | **0.021** | (0.000) |
| | with Additional-data | Content-Boosted CF | 0.149 | 0.228 | 0.019 | 0.000 |
| Model-Based | only UI-matrix | MF | 0.121 | 0.184 | 0.018 | 0.000 |
| | with Additional-data | MF（with genres） | 0.118 | 0.178 | 0.020 | 0.000 |
| | | TWA | 0.017 | 0.017 | 0.017 | **0.023** |
| 深層学習応用 | only UI-matrix | DFM | 0.197 | 0.313 | 0.001 | (0.000) |
| | with Additional-data | DFM（with genres） | 0.192 | 0.305 | 0.001 | 0.000 |

tail のユーザ ID の場合も，全体精度はランキング表示が最も良い．

| 【new ユーザ】 | | | アイテム種別 | | | |
|---|---|---|---|---|---|---|
| 対照手法 | only UI-matrix | ランキング | **0.323** | **0.407** | 0.000 | 0.000 |
| Memory-Based | only UI-matrix | Item-Based | (0.010) | (0.010) | (0.010) | (0.011) |
| | | User-Based | (0.010) | (0.010) | (0.010) | **(0.011)** |
| | with Additional-data | Content-Boosted CF | (0.010) | (0.010) | (0.010) | **(0.011)** |
| Model-Based | only UI-matrix | MF | 0.252 | 0.317 | 0.011 | (0.000) |
| | with Additional-data | MF（with genres） | 0.246 | 0.308 | **0.013** | 0.000 |
| | | TWA | (0.011) | (0.012) | (0.010) | (0.009) |
| 深層学習応用 | only UI-matrix | DFM | 0.260 | 0.327 | 0.000 | (0.000) |
| | with Additional-data | DFM（with genres） | 0.252 | 0.317 | 0.000 | 0.000 |

ユーザ ID が未学習の new なので予測不可能なセグメントが多くなる．new のユーザ ID の場合も，全体的な精度はランキング表示が最も良い．ユーザの種別によらず深層学習応用はランキングに近い性能を示している．

表 6 N = 10 における多様性の評価結果
**Table 6** Results of diversity evaluation for N = 10.

| 手法カテゴリ | 利用するデータ | 手法名 | ENTROPY |
|---|---|---|---|
| 対照手法 | only UI-matrix | ランダム | **8.337** |
| | | ランキング | 2.303 |
| Memory-Based | only UI-matrix | Item-Based | 5.099 |
| | | User-Based | 4.485 |
| | with Additional-data | Content-Boosted CF | 2.841 |
| Model-Based | only UI-matrix | MF | 4.395 |
| | with Additional-data | MF（with genres） | 4.394 |
| | | TWA | 3.635 |
| 深層学習応用 | only UI-matrix | DFM | 2.675 |
| | with Additional-data | DFM（with genres） | 2.533 |

表示機会の分配性をエントロピーで比較した．ランダム表示が最も大きな値となり，どのユーザにも同じアイテムリストを返却するランキング表示が最低値となった．各レコメンド手法はその中間の値に位置している．

分に大きく，機能している．本評価では，Additional Data として比較的単純な映画ジャンルだけを用いたことも加味すると，TWA は提案論文の主張どおり new のアイテム表示に有効な手法である．

Content-Boosted CF 法は，ユーザが head あるいは tail であれば，ランキング以外の手法の中では最も高い性能となり，この手法の提案論文による「疎に強い性質がある」という主張が確認できた．一方で，最も疎の状況である tail × tail では後述する User-Based，Item-Based の方が精度は高い．

Item-Based と User-Based は，ほとんどの状況においてランダムと同水準の 0.01 程度の精度であり，その推薦は機能していない．しかし，tail × tail の状況で，User-Based が最高の精度となった．つまり，ユーザとアイテムの双方がマイナーな場合で高い性能を発揮する傾向にあるといえる．この原因は User-Based の推薦モデルの性質によるものだと考えられる．たとえば，tail アイテムのみを好む tail ユーザが存在するとする．このようなユーザは特徴的なレビュー履歴を持ち，User-Based はそれと類似するユーザを選出するため，tail × tail の状況で優位な推薦を行えると考えられる．

### 4.6 多様性の評価結果

30 日間のデータを学習し，ランダムで抽出した 1,000 人のユーザ ID についての Top10 リストのエントロピーを計算した結果を比較した（表 6）．当然ではあるが，ランダム表示が最も高く 8.337 となる．その一方で，ランキング表示では人気アイテムがすべての表示機会を独占するため，最低値の 2.303 となった．

対して，各レコメンド手法のエントロピーは，2.533 から 5.099 の間にばらついており，ランキング表示に比べて表示機会を分配させていた．このことから，レコメンド手法はランキングよりも表示精度が劣るが，多様性は優れるといえる．ただし，レコメンド手法の中でも推薦精度に優れる DFM と Content-Boosted のエントロピーは他のレコメンド手法に比べてかなり低かった．

## 5. 議論

既存の研究では Cold-Start 問題はデータの少ないユーザとアイテムの推薦課題と定義されている．しかし，本研究の評価結果から，映画アプリケーションにおいては「new ユーザへの推薦は精度の観点からはむしろ容易であり，head ユーザほど推薦が困難になる」という新たな側面が確認された．一方で，アイテムに関しては従来の Cold-Start 問題通りに，new になるほど推薦が困難になる傾向にある．

実験結果ではランキング表示の場合はユーザが，head，tail，new へとデータが少なくなるほど表示精度が改善している．この原因として，new のユーザにとっては，前 30 日間の人気アイテムであっても未知のアイテムであり，まだ魅力的である可能性が高いことがあると考えられる．一方で，head のユーザにとって人気アイテムはすでに既知であり，相対的にその魅力度は下がると考えられる．映画のデータを用いた多くの既存研究ではユーザ側の Cold-Start 問題を評価しているが，それはむしろ容易な推薦セグメントであり，本質的に重要なのはアイテム側の Cold-Start 問題であることを，提案する評価方法によって指摘できた．

このような Cold-Start 問題の新たな側面は，映画，音楽，本などのアプリケーションでも同様に発生しうると考えられる．一方で，ニュースや広告配信などのシステムでは，新規アイテムの配信が特に重要であるため，Cold-Start 問題のまた違った側面が予想される．それらの検証については，本研究では適切なデータを入手できなかったため，今後の課題とした．

## 6. まとめ

本研究では，推薦精度と多様性の双方を考慮したレコメンデーションのオフライン評価方法を提案した．提案する評価方法の特徴は下記の 2 つである．

- ユーザとアイテムを履歴データでの頻度から [new（新規），tail（低頻度），head（高頻度）] に分け，未来に対する推薦精度を比較し，
- 表示機会をアイテムごとに分配する度合い（多様性）を比較する．

MovieLens の映画レビューログデータに本評価法を適応した結果，以下のことが判明した．

- 映画アプリケーションでの Cold-Start 問題は，new や tail のユーザには発生せず，アイテム側のみに発生している．
- ランキングによる推薦は多様性が低いため，ユーザに同じようなアイテムが表示され続ける問題が発生する．多様性に優れる Item-Based や MF 法などを組み合せることが重要である．

一般的に Cold-Start 問題では過去データが少ないほど推薦精度が悪化するとされているが，上記結果はユーザに関しては逆であることを示している．またレコメンド手法が単純なランキングより推薦精度の高い条件を明らかにし，多様性の観点からもランキングに勝ることを明らかにした．これらの結果はレコメンド手法だけでなく，その評価方法自体の研究の重要性を示している．

**謝辞** 本研究は JSPS 科研費（19H04165）の助成を受けたものです．

### 参考文献


[1] 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).

[2] Cremonesi, P., Koren, Y. and Turrin, R.: Performance of recommender algorithms on top-n recommendation tasks, *Proc. 4th ACM Conference on Recommender Systems*, pp.39–46, ACM (2010).

[3] Koren, Y.: Factorization meets the neighborhood: A multifaceted collaborative filtering model, *Proc. 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.426–434, ACM (2008).

[4] Ge, M., Delgado-Battenfeld, C. and Jannach, D.: Beyond accuracy: Evaluating recommender systems by coverage and serendipity, *Proc. 4th ACM Conference on Recommender Systems*, pp.257–260, ACM (2010).

[5] Ziegler, C.-N., McNee, S.M., Konstan, J.A. and Lausen, G.: Improving recommendation lists through topic diversification, *Proc. 14th International Conference on World Wide Web*, pp.22–32, ACM (2005).

[6] Lika, B., Kolomvatsos, K. and Hadjiefthymiades, S.: Facing the cold start problem in recommender systems, *Expert Systems with Applications*, Vol.41, No.4, pp.2065–2073 (2014).

[7] Lathia, N.K.: Evaluating collaborative filtering over time, PhD Thesis, University College London (2010).

[8] Salton, G.: Automatic text processing: The transformation, analysis, and retrieval of information by computer, *Reading: Addison-Wesley*, Vol.169 (1989).

[9] Popescul, A., Pennock, D.M. and Lawrence, S.: Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments, *Proc. 17th Conference on Uncertainty in Artificial Intelligence*, pp.437–444, Morgan Kaufmann Publishers Inc. (2001).

[10] Zhang, M., Tang, J., Zhang, X. and Xue, X.: Addressing cold start in recommender systems: A semi-supervised co-training algorithm, *Proc. 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp.73–82, ACM (2014).

[11] Schein, A.I., Popescul, A., Ungar, L.H. and Pennock, D.M.: Methods and metrics for cold-start recommendations, *Proc. 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.253–260, ACM (2002).

[12] Singh, A.P. and Gordon, G.J.: Relational learning via collective matrix factorization, *Proc. 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.650–658, ACM (2008).

[13] Shi, Y., Larson, M. and Hanjalic, A.: Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges, *ACM Computing Surveys* (*CSUR*), Vol.47, No.1, Article No.3 (2014).

[14] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J.: GroupLens: An open architecture for collaborative filtering of netnews, *Proc. 1994 ACM Conference on Computer Supported Cooperative Work*, pp.175–186, ACM (1994).

[15] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J.: Item-based collaborative filtering recommendation algorithms, *Proc. 10th International Conference on World Wide Web*, pp.285–295, ACM (2001).

[16] Melville, P., Mooney, R.J. and Nagarajan, R.: Content-boosted collaborative filtering for improved recommendations, *AAAI/IAAI*, Vol.23, pp.187–192 (2002).

[17] Koren, Y., Bell, R. and Volinsky, C.: Matrix factorization techniques for recommender systems, *Computer*, Vol.42, No.8, pp.30–37 (2009).

[18] Guo, H., Tang, R., Ye, Y., Li, Z. and He, X.: DeepFM: A factorization-machine based neural network for CTR prediction, arXiv preprint arXiv:1703.04247 (2017).

[19] Park, S.-T., Pennock, D., Madani, O., Good, N. and DeCoste, D.: Naïve filterbots for robust cold-start recommendations, *Proc. 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.699–705 (2006).

[20] Kula, M.: Metadata embeddings for user and item cold-start recommendations, arXiv preprint arXiv:1507.08439 (2015).

[21] Park, S.-T. and Chu, W.: Pairwise preference regression for cold-start recommendation, *Proc. 3rd ACM Conference on Recommender Systems*, pp.21–28 (2009).

[22] Harper, F.M. and Konstan, J.A.: The MovieLens datasets: History and context, *ACM Trans. Interactive Intelligent Systems* (*TIIS*), Vol.5, No.4, Article No.19 (2016).

[23] Hug, N.: Surprise, a Python library for recommender systems (2017), available from ⟨http://surpriselib.com⟩.

[24] Cheng: tensorflow-DeepFM (2018), available from ⟨https://github.com/ChenglongChen/tensorflow-DeepFM⟩.
[25] He, X., Liao, L., Zhang, H., Nie, L., Hu, X. and Chua, T.-S.: Neural collaborative filtering, *Proc. 26th International Conference on World Wide Web*, pp.173–182, International World Wide Web Conferences Steering Committee (2017).
[26] Wang, X., He, X., Cao, Y., Liu, M. and Chua, T.-S.: KGAT: Knowledge graph attention network for recommendation, *Proc. 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp.950–958 (2019).

**内田 匠**

2009年（株）セプテーニ入社．2015年（株）リクルート・コミュニケーションズ入社．2020年（株）つくばAIカツヨウシエンセンター設立．2020年インキュデータ（株）入社．2015年筑波大学大学院ビジネス科学研究科博士後期課程で研究活動．

**吉田 健一** （正会員）

1980年（株）日立製作所入社．1992年工学博士（大阪大学）．2002年より筑波大学教授．インターネットと機械学習技術の応用研究に従事．