

内容情報を活用した推薦システムのためのメタ学習

三角 周平[†] 田中 佑典^{††} 岩田 具治^{††}

[†] 奈良先端科学技術大学院大学 先端科学技術研究科 〒630-0192 奈良県生駒市高山町 8916-5

^{††} NTT コミュニケーション科学基礎研究所 〒619-0237 京都府相楽郡精華町光台 2-4 NTT 京阪奈ビル

E-mail: [†]misumi.shuhei.mr4@is.naist.jp, ^{††}{yusuke.tanaka.rh,tomoharu.iwata.gy}@hco.ntt.co.jp

あらまし 推薦システムにおけるメタ学習の活用は、ユーザーや商品の評価値が少ないデータに対する機械学習手法のアプローチの一つとして注目されており、ユーザーや商品の評価値の少ないデータを持つサービスでの推薦予測精度向上が期待できる。しかし商品の評価値のみのデータでは、欠損値が非常に多い場合に大幅に精度が落ちてしまうことが懸念されている。そこで本研究ではユーザーによる商品の評価に、年齢、職業、性別といったユーザーの内容情報や、ジャンル、リリース日といった商品の内容情報を埋め込み、ユーザーと商品の相互関係を活用できるメタ学習を用いた推薦手法を提案する。提案手法は評価値のみ入力する手法と比べて、評価値予測精度が高いことを示す。

キーワード 推薦システム, ハイブリッド型推薦システム, メタ学習, 深層学習, 行列分解

1 はじめに

「推薦システム (recommender system)」とは、ユーザーの行動にまつわる情報 (商品の購買履歴等) に基づいて、ユーザーにとって有用であると思われる対象 (商品, 情報等) を選出し、それらをユーザーの目的に沿った形で提供するシステムを指す [15] [16]。推薦システムは現在ネットショッピングを代表とする e コマース事業やソーシャルメディア, 広告といった様々な場面で必要不可欠な技術となっている。その背景として、情報化技術の進展に伴い、個人や団体が容易かつ低コストで発信可能になったことで、大量の情報の蓄積や流通が我々の生活にとって身近になっていることが挙げられる [15] [16]。

今日の推薦システムでは、ビッグデータの活用によって、ユーザーの属性情報や他の商品の概要等の豊富な情報を十分に用いての商品の推薦及び未知の商品の評価値予測が可能となっている [12]。その一方で、新規に設立されたばかりのサービスでは、ユーザーや商品の評価値のデータが少なく、ユーザーと関連する商品の傾向を掴みにくいといった問題がある。

こうした少ないデータに対応できる機械学習手法として、メタ学習 [4] [7] [3] に着目する。メタ学習とは、「学習することを学習する (learning-to-learn)」プロセスを指し、少数データであっても新しいタスクに適応するように学習の仕方を学習することができる [4] [7] [3]。メタ学習を推薦システムに応用すれば、各ユーザーによる商品の評価値が少ないデータであっても、ユーザーに関連した商品の傾向を掴む方法を学習することで、よりユーザーの嗜好に沿った推薦や評価値の予測を行うことができると考えられている。

メタ学習を用いた推薦システムの先行研究 [8] では、各ユーザーによる各商品の評価のデータを複数用いて、複数の推薦システムのタスクを作り出し、タスクごとにモデルの学習やテストをシミュレートすることで、各タスクにおける学習プロセスを通じてモデルを学習している。これにより、少数のデータを持

つ未知のタスクにおける評価値の予測精度向上に寄与している。しかし、こうした学習アルゴリズムでもデータ内の評価が極端に少なければ、評価値の予測が困難になる。

そこで、本研究ではユーザーによる商品の評価に、年齢、職業、性別といったユーザーの内容情報や、ジャンル、リリース日といった商品の内容情報を埋め込み、ユーザーと商品の相互関係を活用できるメタ学習を用いた推薦手法を提案する。評価が極端に少ないデータであっても、ユーザーと商品の相互関係を補助情報として活用することができるため、未知のタスクにおける評価値の予測精度の向上が見込まれる。

2 関連研究

2.1 ハイブリッド型推薦システム

内容ベースフィルタリングと協調フィルタリングを組み合わせ、これらの短所を補うことを可能としたハイブリッド型推薦システムが提案されている [13] [18]。協調フィルタリングでは、新しいユーザーや商品に関するレーティングが弱点であった [11] [13] [18]。一方で内容ベースフィルタリングではユーザーや商品の特徴を用いることができるため、新しい商品のレーティングに役立てることができる [13] [17]。協調フィルタリングに内容ベースフィルタリングを組み込むことにより、ユーザーにとって意外な商品の推薦を行いつつ、新しい未知の商品のレーティングが可能となるので、ユーザーにとって推薦出来る商品の幅を広げることができる [18]。本研究では、各ユーザーによる各商品の評価値行列に基づいて行列分解されたユーザーと商品のベクトルに各ユーザーや商品の内容情報を埋め込んだハイブリッド型推薦システムを具現化している。

2.2 メタ学習を用いた推薦システム

商品の大半が未知の新しいものであった時、すなわち評価値の欠損が大半を占めるスパースデータに対する推薦システムについても研究が進められている。本項目では本研究の趣旨と同

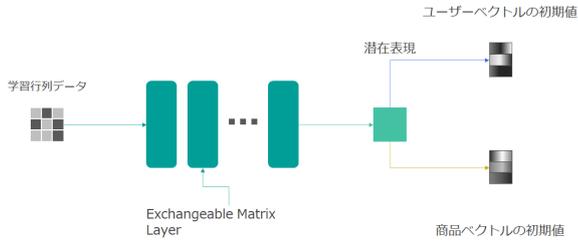


図1 Exchangeable Matrix Layer を用いた予測モデル及び予測モデルから出力された潜在表現からユーザーベクトルの初期値と商品ベクトルの初期値の算出

じく、スパースデータに対処する意図でメタ学習を用いた推薦システム [14] [2] について主に注目する。メタ学習を用いることで、既存の推薦システムのタスクでメタ学習されたパラメータを用いてスパースデータの予測精度向上に貢献できるとされる。特に少数の勾配更新を駆使して、新規ユーザーに対する商品のレーティングの汎用性を高めたメタ学習アーキテクチャ [2] や毎回更新される新しい商品に対する推薦の推薦システムに対するアプローチ [14] 等が提案されている。本研究のベースとなったメタ学習手法 [8] では、ユーザーや商品の数が互いに異なるタスクを入力することが可能であり、ユーザーや商品の関係性の使用や異なる数のユーザー及び商品データ間での行列分解を実現している。なお本研究では、行列分解して得られたユーザー及び商品の特徴ベクトルにそれぞれユーザーと商品の内容情報を埋め込むことによって、データ内の評価が極端に少ないスパースデータにも対応できるようにしている。

3 提案手法

3.1 問題設定

はじめに提案手法の説明に用いる変数について定義する。meta-training の段階で用いる D 個のタスクのうち、タスク d において、 N_d 人のユーザーによる M_d 個の商品のレーティングを行列 $\mathbf{X}_d \in \mathbb{R}^{N_d \times M_d}$ とし、 \mathbf{X}_d から作成された学習行列データを $\mathbf{X} \in \mathbb{R}^{N \times M}$ とする。meta-test の段階で用いるタスクでの、ユーザー数 N_* による M_* 個の商品のレーティングを行列 $\mathbf{X}_* \in \mathbb{R}^{N_* \times M_*}$ とする。行列 \mathbf{X}_* は meta-training で使用しない。以上に加え、 \mathbf{X}_d に関連する L^{user} 次元の特徴を持つユーザーの内容情報ベクトル $\mathbf{a}_{dn}^{\text{user}} \in \mathbb{R}^{L^{\text{user}}}$ 、 L^{item} 次元の特徴を持つ商品の内容情報ベクトル $\mathbf{a}_{dm}^{\text{item}} \in \mathbb{R}^{L^{\text{item}}}$ を導入する。本論文では問題設定として、meta-test 段階で未知のタスクの行列 \mathbf{X}_* と \mathbf{X}_* に関連する各ユーザーや商品の内容情報 $\mathbf{a}_{*n}^{\text{user}}$ 、 $\mathbf{a}_{*m}^{\text{item}}$ を入力として与え、 \mathbf{X}_* 上の欠損値の予測精度の向上を目的とする。

3.2 モデル

3.2.1 評価値行列の埋め込みに使用するモデル

本研究では、あるタスクで使用されるレーティングの行列からサンプリングして得られた学習行列データを主に入力として、予測モデルから潜在変数 z_{nm} を出力する。予測モデルでは多様

多様な行列に対して適切にパラメータ調整できるようにするため、入力層、中間層、出力層にそれぞれ Exchangeable Matrix Layer [6] を用いている (図 1 参照)。Exchangeable Matrix Layer では式 (1) のような処理を行う。

$$z_{nm}^{(l+1)} = \sigma \left(w_1^{(l)} z_{nm}^{(l)} + \frac{w_2^{(l)}}{N} \sum_{n'=1}^N z_{n'm}^{(l)} + \frac{w_3^{(l)}}{M} \sum_{m'=1}^M z_{nm'}^{(l)} + \frac{w_4^{(l)}}{NM} \sum_{n'=1}^N \sum_{m'=1}^M z_{n'm'}^{(l)} + w_5^{(l)} \right) \quad (1)$$

$z_{nm}^{(l)} \in \mathbb{R}$ は l 層目 (n, m) に位置する表現ベクトルの要素を表す。 $w_i^{(l)} \in \mathbb{R}$ は l 層目の重み付けパラメータである。そして、 $\sigma(\cdot)$ は活性化関数である。最初の層では、 $N \times M$ 行列 $\mathbf{X} \in \mathbb{R}^{N \times M}$ が入力として与えられた時の (n, m) 番目の要素 $x_{nm} \in \mathbb{R}$ が与えられる、すなわち $z_{nm}^{(0)} = x_{nm}$ が与えられる。最後の層では活性化関数 $\sigma(\cdot)$ は導入されない。

3.2.2 内容情報の埋め込みに使用するモデル

潜在変数 z_{nm} から各商品、各ユーザーの平均を求め、各 Feed-forward Neural Network $f_{\mathbf{u}}$ 、 $f_{\mathbf{v}}$ を用いてユーザーベクトルの初期値 $\mathbf{u}_n^{(0)}$ と商品ベクトルの初期値 $\mathbf{v}_m^{(0)}$ を算出する (式 2, 3, 図 1 参照)。

$$\mathbf{u}_n^{(0)} = f_{\mathbf{u}} \left(\frac{1}{M} \sum_{m=1}^M z_{nm} \right) \quad (2)$$

$$\mathbf{v}_m^{(0)} = f_{\mathbf{v}} \left(\frac{1}{N} \sum_{n=1}^N z_{nm} \right) \quad (3)$$

その後ユーザーの内容情報ベクトル $\mathbf{a}_n^{\text{user}}$ を Feed-forward Neural Network $h_{\mathbf{A}_u} : \mathbb{R}^{L_u} \rightarrow \mathbb{R}^{L_u}$ 、商品の内容情報ベクトル $\mathbf{a}_m^{\text{item}}$ を Feed-forward Neural Network $h_{\mathbf{A}_v} : \mathbb{R}^{L_v} \rightarrow \mathbb{R}^{L_v}$ に入力し、ユーザーの内容情報ベクトルの潜在表現 $\mathbf{z}_{\mathbf{A}_u, n}$ と商品の内容情報ベクトルの潜在表現 $\mathbf{z}_{\mathbf{A}_v, m}$ を出力する (式 (4), (5) 参照, 図 2 参照)。

$$\mathbf{z}_{\mathbf{A}_u, n} = h_{\mathbf{A}_u}(\mathbf{a}_n^{\text{user}}) \quad (4)$$

$$\mathbf{z}_{\mathbf{A}_v, m} = h_{\mathbf{A}_v}(\mathbf{a}_m^{\text{item}}) \quad (5)$$

3.2.3 内容情報との連結に使用するモデル

ユーザー及び商品ベクトルの初期値を各内容情報ベクトルの潜在表現とそれぞれ列方向に連結 (concatenation) させる (図 2 参照)。式 (4) から算出されたユーザーベクトルの初期値 $\mathbf{u}_n^{(0)}$ とユーザーの内容情報ベクトルの潜在表現 $\mathbf{z}_{\mathbf{A}_u, n}$ の concatenation ($\mathbf{u}_n^{(0)} \parallel \mathbf{z}_{\mathbf{A}_u, n}$) を Feed-forward Neural Network $h_{\mathbf{u}} : \mathbb{R}^{K+L_u} \rightarrow \mathbb{R}^K$ に入力し、内容情報を埋め込んだユーザーベクトル $\mathbf{u}_n^{(0)}$ を出力する (式 (6), 図 2 参照)。同様に、式 (5) から算出された商品ベクトルの初期値 $\mathbf{v}_m^{(0)}$ と商品の内容情報ベクトルの潜在表現 $\mathbf{z}_{\mathbf{A}_v, m}$ の concatenation ($\mathbf{v}_m^{(0)} \parallel \mathbf{z}_{\mathbf{A}_v, m}$) を Feed-forward Neural Network $h_{\mathbf{v}} : \mathbb{R}^{K+L_v} \rightarrow \mathbb{R}^K$ に入力し、内容情報を埋め込んだ商品ベクトル $\mathbf{v}_m^{(0)}$ を出力する (式 (7) 参照)。ここで式 (6) と (7) において、concatenation を表す記号

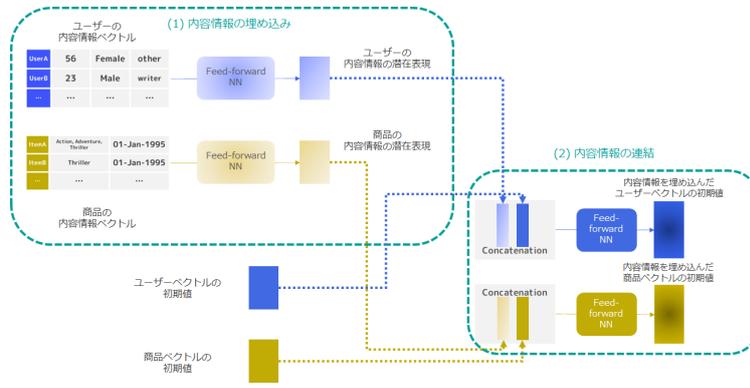


図2 内容情報の埋め込みと連結

として $++$ を用いている.

$$\mathbf{u}_n^{(0)} = h_{\mathbf{U}}(\mathbf{u}_n^{(0)} ++ \mathbf{z}_{\mathbf{A}_u, n}) \quad (6)$$

$$\mathbf{v}_m^{(0)} = h_{\mathbf{V}}(\mathbf{v}_m^{(0)} ++ \mathbf{z}_{\mathbf{A}_v, m}) \quad (7)$$

3.2.4 ユーザーベクトルと商品ベクトルの更新

内容情報を埋め込んだユーザーベクトル $\mathbf{u}_n^{(0)}$ と内容情報を埋め込んだ商品ベクトル $\mathbf{v}_m^{(0)}$ を用いて、ユーザーベクトル $\mathbf{u}_n^{(t)} \in \mathbb{R}^{N \times K}$ と商品のベクトル $\mathbf{v}_m^{(t)} \in \mathbb{R}^{N \times K}$ をそれぞれ更新する (図3参照). その際、ユーザーベクトル $\mathbf{u}_n^{(t)}$ と商品のベクトル $\mathbf{v}_m^{(t)}$ の内積を用いて損失を計算する. 計算式は式 (8) の通りに表される.

$$\mathcal{L} \propto - \sum_{n=1}^N \sum_{m=1}^M b_{nm} (\mathbf{u}_n^{(t)\top} \mathbf{v}_m^{(t)} - x_{nm})^2 \quad (8)$$

加えて、先行研究では予測モデルの入力データに用いた学習行列データ \mathbf{X} の性質がユーザーベクトル $\mathbf{u}_n^{(t)}$ と商品のベクトル $\mathbf{v}_m^{(t)}$ により反映されるようにするため、式 (2) のユーザーベクトル $\mathbf{u}_n^{(0)}$ と式 (3) 商品のベクトルを用いて $\mathbf{v}_m^{(0)}$ 式 (9) の損失を導入する.

$$\mathcal{L}_{\text{fit}} \propto - \sum_{n=1}^N \|\mathbf{u}_n^{(t)} - \mathbf{u}_n^{(0)}\|^2 + \sum_{m=1}^M \|\mathbf{v}_m^{(t)} - \mathbf{v}_m^{(0)}\|^2 \quad (9)$$

損失 \mathcal{L} と \mathcal{L}_{fit} を用いて、正則化パラメータを λ , ユーザーベクトル及び商品ベクトルの学習率 η を $\eta > 0$ とし、ユーザーベクトル $\mathbf{u}_n^{(t)}$ と商品ベクトル $\mathbf{v}_m^{(t)}$ の更新を行う (式 (10), (11) 参照). 各更新は T 回行われる.

$$\begin{aligned} \mathbf{u}_n^{(t+1)} &= \mathbf{u}_n^{(t)} - \eta \frac{\partial}{\partial \mathbf{u}_n^{(t)}} (\mathcal{L} + \lambda \mathcal{L}_{\text{fit}}) \\ &= \mathbf{u}_n^{(t)} - \eta \left(\sum_{m=1}^M b_{nm} (\mathbf{u}_n^{(t)\top} \mathbf{v}_m^{(t)} - x_{nm}) \mathbf{v}_m^{(t)} + \lambda (\mathbf{u}_n^{(t)} - \mathbf{u}_n^{(0)}) \right) \end{aligned} \quad (10)$$

$$\begin{aligned} \mathbf{v}_m^{(t+1)} &= \mathbf{v}_m^{(t)} - \eta \frac{\partial}{\partial \mathbf{v}_m^{(t)}} (\mathcal{L} + \lambda \mathcal{L}_{\text{fit}}) \\ &= \mathbf{v}_m^{(t)} - \eta \left(\sum_{n=1}^N b_{nm} (\mathbf{u}_n^{(t)\top} \mathbf{v}_m^{(t)} - x_{nm}) \mathbf{u}_n^{(t)} + \lambda (\mathbf{v}_m^{(t)} - \mathbf{v}_m^{(0)}) \right) \end{aligned} \quad (11)$$

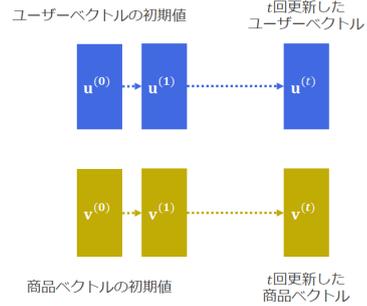


図3 ユーザーベクトルと商品ベクトルの更新

T 回の更新後、評価値行列 $\hat{x}_{nm} = \mathbf{u}_n^{(T)\top} \mathbf{v}_m^{(T)}$ を構成して評価値を予測する.

3.3 メタ学習

メタ学習では、モデルパラメータ Θ , すなわち, Exchangeable Matrix Layer のパラメータ $\{\{w_i^{(l)}\}_{i=1}^5\}_{l=1}^{L-1}$, 各 Feed-forward Neural Network $f_{\mathbf{U}}, f_{\mathbf{V}}, h_{\mathbf{A}_u}, h_{\mathbf{A}_v}, h_{\mathbf{U}}, h_{\mathbf{V}}$ のパラメータ, ユーザーベクトル及び商品ベクトルの更新の際に用いられたハイパーパラメータ λ をそれぞれ学習する. そのために、式 (12) の損失関数を最小化するモデルパラメータ Θ を求める.

$$\mathbb{E}_{\mathbf{X}, \mathbf{B}, \mathbf{X}', \mathbf{B}'} [\|b_{nm} (x'_{nm} - \mathbf{u}_n^\top \mathbf{v}_m)\|^2] \quad (12)$$

\mathbf{X}, \mathbf{X}' は学習行列データ及びテスト行列データを表し、このうちテスト行列データ \mathbf{X}' の要素を x'_{nm} と表している. \mathbf{B}, \mathbf{B}' は学習行列データ及びテスト行列データに対応する各 Binary Indicator Matrix を表しており、このうちテスト行列データ \mathbf{X}' における Binary Indicator Matrix \mathbf{B}' の要素を b'_{nm} と表している.

提案手法におけるメタ学習フレームワークを図4, メタ学習アルゴリズムをアルゴリズム1に示す.

4 評価実験

4.1 データセット

本研究では従来手法で使用された, Movielens (ML100K) [5] のデータセットを用いた. ML100K は 943 人のユーザーによる 1682 個の商品の評価値 100,000 件を有する. なお本研究で

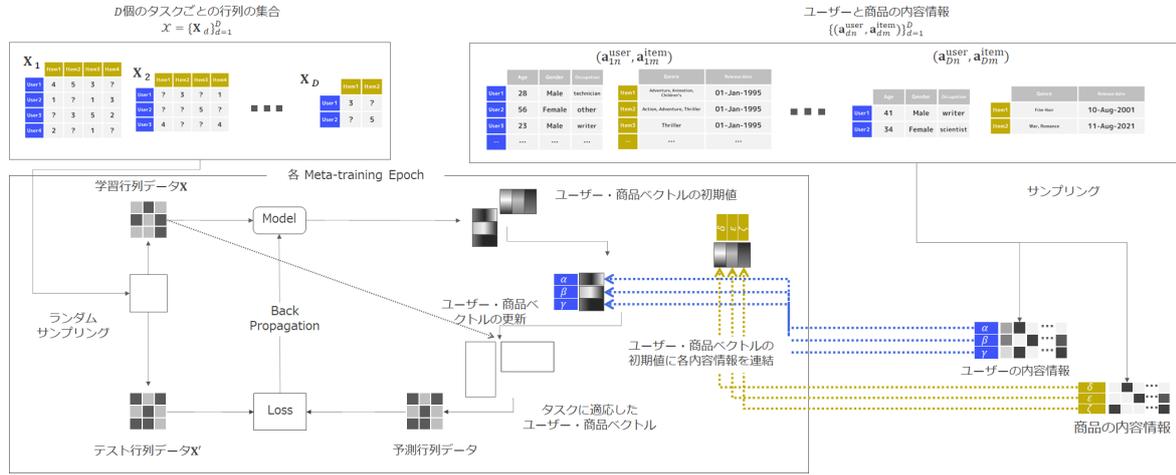


図4 提案手法のフレームワーク

アルゴリズム 1 提案手法のメタ学習アルゴリズム

Input: メタ学習データセット: $\mathcal{X} = \{\mathbf{X}_d\}_{d=1}^D$, ユーザー数 N , 商品数 M , 学習行列データの割合 R , ユーザーベクトル $\mathbf{u}_n^{(t)}$ と商品ベクトル $\mathbf{v}_m^{(t)}$ の更新回数 T , ユーザーと商品の内容情報 $\{(\mathbf{a}_{dn}^{\text{user}}, \mathbf{a}_{dm}^{\text{item}})\}_{d=1}^D$,

Output: 学習後のモデルパラメータ Θ

- 1: モデルパラメータ Θ の初期化
- 2: **while** 収束 **do**
- 3: 無作為にタスク d の選択
- 4: タスク d の行列 \mathbf{X}_d から無作為にユーザー数 N , 商品数 M の行列 \mathbf{X} をサンプリングして作成
- 5: サンプリングされた行列 \mathbf{X} のうち, 割合 R 分を学習行列データ \mathbf{X} , 残りをテスト行列データ \mathbf{X}' に分割
- 6: 学習行列データ \mathbf{X} を用いて予測モデルからユーザーベクトルの初期値 $\mathbf{u}_n^{(0)}$ と商品ベクトルの初期値 $\mathbf{v}_m^{(0)}$ を生成 (式 (2), (3) 参照)
- 7: それぞれ Feed-forward Neural Network $h_{\mathbf{A}_u}$, $h_{\mathbf{A}_v}$ を介して, ユーザーの内容情報ベクトル $\mathbf{a}_{dn}^{\text{user}}$, 商品の内容情報ベクトル $\mathbf{a}_{dm}^{\text{item}}$ をユーザーベクトルの初期値 $\mathbf{u}_n^{(0)}$ と商品ベクトルの初期値 $\mathbf{v}_m^{(0)}$ に埋め込み, Feed-forward Neural Network $h_{\mathbf{U}}$, $h_{\mathbf{V}}$ を介してユーザーベクトルの初期値 $\mathbf{u}_n^{(0)}$ と商品ベクトルの初期値 $\mathbf{v}_m^{(0)}$ を算出する (式 (4), (5), (6), (7) 参照)
- 8: ユーザーベクトル $\mathbf{u}_n^{(0)}$ と商品ベクトル $\mathbf{v}_m^{(0)}$ を各更新式 (10), (11) を基に T 回更新
- 9: 損失関数の計算 (式 (12) 参照)
- 10: 確率的勾配降下法によるパラメータ Θ の更新
- 11: **end while**

は, 評価値を平均 0, 分散が 1 になるように正規化した.

ML100K におけるユーザーの内容情報では, 年齢, 2 パターンの性別 (男女), 21 個の職業, 商品の内容情報では, 19 種類のジャンルとリリース日で構成される. 本研究では以上の項目のうち, 性別と職業, ジャンルを 0 と 1 の One-hot vector で表し, 年齢とリリース日を $[0, 1]$ 間で正規化した.

実験に用いるデータセットのうち, 70% を meta-training, 10% を meta-validation, 残りを meta-test の段階で用いる 3 つのデータセットに分割して実験を行った. meta-test の段階では, 元のデータから無作為に $N \times M$ の行列データを作成し, そのうち半分の評価値を欠損値とした状態でテストを行った.

5 条件設定

本研究で使用されるモデルはノード数 32 の Exchangeable Matrix Layer が 3 層で構成される. Feed-forward Neural Network $f_{\mathbf{U}}$, $f_{\mathbf{V}}$ は中間層と出力層のノード数 32 を持つ 3 層で構成される. 活性化関数は ReLU 関数 $\text{ReLU}(x) = \max(0; x)$ [1] を使用している. 式 (10), (11) のユーザーベクトル及び商品ベクトルの勾配降下ステップによる更新において, 学習率は $\eta = 10^{-2}$, 各ベクトルの更新回数は $T = 10$ とした. モデルは Adam [10] を用いて最適化され, Adam の学習率は 10^{-4} とした. バッチサイズは 16, ドロップアウト率は 0.1, メタ学習のエポック数は 1000 と設定した.

式 (4) でユーザーの内容情報ベクトルの入力に用いられる Feed-forward Neural Network $h_{\mathbf{A}_u}$, 式 (5) で商品の内容情報ベクトルで用いられる Feed-forward Neural Network $h_{\mathbf{A}_v}$ はそれぞれ 3 層で構成され, 入力層のノード数が各内容情報ベクトルの特徴の長さ設定している. なお中間層のノード数 32, 出力層のノード数 32 と設定している. 式 (6) で内容情報を埋め込んだユーザーベクトルを出力するのに用いられる Feed-forward Neural Network $h_{\mathbf{U}}$, 式 (7) で内容情報を埋め込んだ商品ベクトルを出力するのに用いられる Feed-forward Neural Network $h_{\mathbf{V}}$ もそれぞれ入力層のノード数 64, 中間層のノード数 32, 出力層のノード数 32 の 3 層で構成される.

5.1 比較対象

提案手法の有効性を確認するために, 先行研究の手法 [8] とは他に以下の比較手法を用意した.

- MAML (Model-Agnostic Meta-Learning) [4] モデルや損失関数にとらわれないメタ学習手法 [4] [2] であり, ニューラルネットワーク全体に対してタスク毎及び全体でモデルを学習する [4]. 本研究では予測モデルは提案手法と同じくノード数 32 の Exchangeable Matrix Layer が 3 層で構成される.

- EML (Exchangeable Matrix Layer) [8] [6] のみ 先行研究の Exchangeable Matrix Layer を用いたモデルの出力

表 1 提案手法と従来手法との比較結果 (RMSE と標準偏差を示す)

手法	行列サイズ 30	行列サイズ 50	行列サイズ 80
提案手法	0.960 ± 0.047	0.929 ± 0.039	0.913 ± 0.035
先行研究	0.968 ± 0.046	0.942 ± 0.047	0.922 ± 0.034
MAML	0.967 ± 0.043	0.941 ± 0.037	0.922 ± 0.034
EML	0.969 ± 0.046	0.949 ± 0.040	0.930 ± 0.037

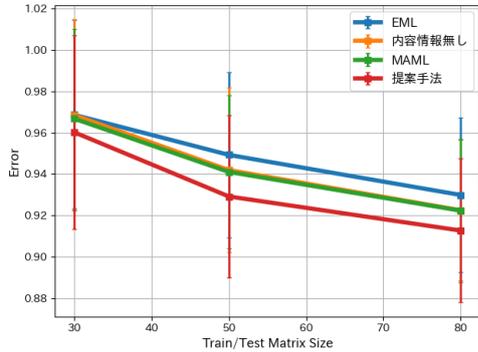


図 5 提案手法と従来手法との比較結果 (RMSE と標準偏差を示す)

において、ユーザーベクトルと商品ベクトルに行列分解しないモデルである。提案手法と同じくノード数 32 の Exchangeable Matrix Layer が 3 層で構成される。

5.2 評価指標

本研究では、評価指標として RMSE (Root Mean Squared Error, 平均二乗誤差の平方根) [9] を meta-test 時におけるクエリデータに導入し、比較手法と誤差を比較することにより提案手法の有効性を測った。RMSE を式 (13) にて示す [9]。 x'_{nm} , \hat{x}_{nm} は、 $N \times M$ のテスト行列データ \mathbf{X}' , 予測行列データ $\hat{\mathbf{X}}$ の (n, m) の要素を表す。

$$\text{RMSE} = \sqrt{\frac{1}{NM} \sum_{n=1}^N \sum_{m=1}^M (x'_{nm} - \hat{x}_{nm})^2} \quad (13)$$

6 実験結果

提案手法と従来手法との比較結果、各内容情報の各特徴間の比較結果について述べる。

6.1 従来手法との比較

図 5 と表 1 で示すように、メタ学習でランダムサンプリングされたどの行列のサイズでも、提案手法が最高精度を記録した。合わせて先行研究と同様行列サイズが大きくなるにつれて、提案手法の精度が高くなっている。

6.2 各特徴間の比較

行列サイズ及び特徴毎の結果を図 6 及び表 2 に示す。まず全体を通して、各行列サイズで商品のリリース日のみの内容情報を使用した時が、提案手法 (年齢+性別+職業+ジャンル+リリース日) より高い精度を記録した。この結果に加えて先行研究と比較して、ユーザーの性別、職業のみの内容情報を使用した時が精度が高く、ユーザーの年齢、商品のジャンルのみの内

表 2 行列サイズ及び使用した内容情報の特徴別の比較結果 (RMSE と標準偏差を示す)

行列サイズ	使用した内容情報の特徴	RMSE ± 標準偏差
30	年齢のみ	0.973 ± 0.034
	性別のみ	0.943 ± 0.040
	職業のみ	0.962 ± 0.075
	ジャンルのみ	0.987 ± 0.057
	リリース日のみ	0.926 ± 0.055
	提案手法	0.960 ± 0.047
内容情報無し		0.968 ± 0.046
50	年齢のみ	0.955 ± 0.037
	性別のみ	0.937 ± 0.031
	職業のみ	0.929 ± 0.049
	ジャンルのみ	0.954 ± 0.038
	リリース日のみ	0.920 ± 0.046
	提案手法	0.929 ± 0.039
内容情報無し		0.942 ± 0.040
80	年齢のみ	0.920 ± 0.030
	性別のみ	0.916 ± 0.030
	職業のみ	0.920 ± 0.037
	ジャンルのみ	0.936 ± 0.028
	リリース日のみ	0.910 ± 0.035
	提案手法	0.913 ± 0.035
内容情報無し		0.922 ± 0.034

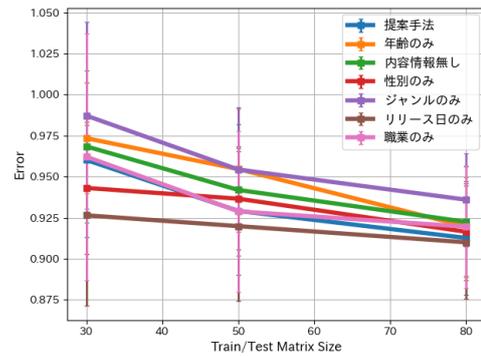


図 6 行列サイズ及び使用した内容情報の特徴別の比較結果 (RMSE と標準偏差を示す)

容情報を使用した時が精度が低くなっている。

7 考察

7.1 従来手法との比較

実験結果における図 5 と表 1 を踏まえて以下考察を述べる。実験結果より、提案手法が各行列サイズでの最高精度を記録したことについて、提案手法で予測モデルから出力されたユーザーベクトルと商品ベクトルに各内容情報を補助的に活用することは、先行研究で用いられたモデルの学習を維持し、その後のユーザーベクトルと商品ベクトルの更新に良い影響を与えていると言える。その上で小さい行列サイズでも先行研究を上回った精度を記録したことから、内容情報を用いることで、ユーザーまたはユーザーによる商品のレーティングが疎らな場合でも、ユーザーと商品の関係性を考慮した推薦の実現に貢献していると考えられる。

7.2 各特徴間の比較

実験結果における図6及び表2を踏まえて以下考察を述べる。実験結果より、他の特徴と比べて、本研究で用いたユーザーの性別や職業、商品のリリース日はそれぞれユーザーの嗜好と相関性があり、未知の商品のレーティングにもその傾向が反映できていると考えられる。

一方で、ユーザーの年齢や商品のジャンルのみを内容情報として用いた時の実験結果の考察として、これらの特徴がユーザー・商品間との相関性とミスマッチを起こしたことが考えられる。すなわち、ユーザーの年齢や商品のジャンルとユーザーの嗜好に関係性が無いことが推測できる。さらに商品のジャンルにおいて、他のユーザーの性別、職業といったOne-hot vector化して使用した特徴とは異なり、複数のジャンルを有している(ユーザーの性別は男女の2パターンでユーザーは必ずどちらか片方を有する、職業も21個のうち必ず1つのみを有する)ため、特徴が複雑化して精度低下に繋がったと推測できる。

8 まとめ

本研究ではユーザーによる商品の評価のみの情報だけでなく、ユーザーの内容情報や、商品の内容情報を活用して、ユーザーと商品の相互関係を活用できる推薦システム手法を提案した。先行研究では協調フィルタリングをベースとしているため、未知の商品のレーティングに課題点を含んでいたが、本研究で内容ベースフィルタリングをベースとした方法として内容情報を活用することによって、未知の商品の評価予測の精度を高められたことを確認できた。

今後の検討事項として、内容情報に多様な特徴表現を用いてそれに適応可能なモデル構築を行うことが挙げられる。例えば、ユーザーや商品について説明している文をWord2vec等を通じて数値化して内容情報として用いるといった案や、内容情報の時系列を考慮したフレームワークを作成するといった案が考えられる。こうした研究により、提案手法がより実用的になることを期待したい。

文 献

- [1] Abien Fred Agarap. Deep learning using rectified linear units (relu), 2018.
- [2] Homanga Bharadhwaj. Meta-learning for user cold-start recommendation. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2019.
- [3] 田口健太郎 Denis Pastory. メタ学習によるニューラルネットワークの感度解析. In *人工知能学会全国大会論文集 第34回 (2020)*, pages 1J4GS201–1J4GS201. 一般社団法人 人工知能学会, 2020.
- [4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 06–11 Aug 2017.
- [5] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.
- [6] Jason S. Hartford, Devon R. Graham, Kevin Leyton-Brown,

- and Siamak Ravanbakhsh. Deep models of interactions across sets. In *International Conference on Machine Learning*, 2018.
- [7] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5149–5169, 2022.
 - [8] Tomoharu Iwata. Meta-learning for matrix factorization without shared rows or columns. *arXiv preprint arXiv:2106.15133*, 2021.
 - [9] Sambandam Jayalakshmi, Narayanan Ganesh, Robert Čep, and Janakiraman Senthil Murugan. Movie recommender systems: Concepts, methods, challenges, and future directions. *Sensors*, 22(13), 2022.
 - [10] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.
 - [11] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
 - [12] Yuanzhe Peng. A survey on modern recommendation system based on big data, 2022.
 - [13] Francesco Ricci, Lior Rokach, and Bracha Shapira. *Recommender Systems Handbook*. Springer, 2015.
 - [14] Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. A meta-learning perspective on cold-start recommendations for items. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, pages 6907–6917, Red Hook, NY, USA, 2017. Curran Associates Inc.
 - [15] 神島敏弘. 推薦システム-情報過多時代をのりきる (<特集>情報のフィルタリング). *情報の科学と技術*, 56(10):452–457, 2006.
 - [16] 神島敏弘. 推薦システムのアルゴリズム (1). *人工知能*, 22(6):826–837, 2007.
 - [17] 神島敏弘. 推薦システムのアルゴリズム (2). *人工知能*, 23(1):89–103, 2008.
 - [18] 神島敏弘. 推薦システムのアルゴリズム (3). *人工知能*, 23(2):248–263, 2008.