

# 評価傾向の差異を考慮した分散表現による協調フィルタリング

Collaborative filtering based on distributed expression considering differences in evaluation trends

\*1後藤 亮介      藤波 英輝      楊 添翔      後藤 正幸  
Ryousuke Goto    Hideki Fujinami    Tianxiang Yang    Masayuki Goto

早稲田大学理工学術院

Faculty of Science and Engineering, Waseda University

Recently, the preferences of users using EC sites are diversifying. Therefore, a recommender system that grasps the preferences of each user and recommends items matching with the preferences became more and more important in business activities. In the conventional recommendation proposed by Kuzmin, the items are divided into two groups depending on whether or not a user highly evaluated each item. Then, the same item group is regarded as one purchase series and learned by a single model. However, if items with different evaluation tendencies are learned by a single model, evaluation tendencies for items can not be properly expressed. In this research, we propose a recommender system that reflects users' preferences by expressing items with different two evaluation trends by different models. Finally, by applying the proposed method to the movie evaluation history data and comparing it with the conventional method, we show the effectiveness of our proposal.

## 1. 研究背景と目的

近年、情報技術の発展に伴い、EC サイトが取り扱うアイテム数が増加し、利用するユーザの嗜好も多様化している。そのため、EC サイトを運営する企業にとって、過去の購買履歴データから各ユーザの嗜好を把握し、嗜好に合致したアイテムを提示するための推薦システムは重要な Web マーケティング技術の 1 つになっている。また様々な EC サイト上では、評価履歴データという形で、ユーザが購買したアイテムに対して満足したか否かを判断できる評価値が導入されている。

推薦手法の 1 つとして、嗜好が似ているユーザに同じものを推薦する協調フィルタリング [神島 2003] がある。協調フィルタリングの中には、評価履歴データを用いて学習し、未評価アイテムの中で予測評価値の高いアイテムを推薦するアイテムベース協調フィルタリングがある。評価履歴が付いていない購買履歴データを用いて学習した場合、購入したアイテムにユーザが満足しているかどうか分からない。しかし、評価履歴データを用いることで、選択したアイテムが好きか嫌いかを考慮することができるため、ユーザの嗜好をより反映した推薦システムを実現することができる。評価履歴データを用いたアイテムベース協調フィルタリング手法の 1 つに、Kuzmin の手法 [Kuzmin 2017] がある。この手法は、Item2Vec [Barkan 2016] をベースとした未評価アイテムの評価値予測モデルであり、Item2Vec に入力する学習データに対する前処理として、評価値をもとにデータを分割している。過去に評価を付けた全アイテムに対する評価値の平均値をユーザ毎に算出し、平均値以上の評価値が付与されたアイテムを“高評価アイテム群”，平均値未満の評価値が付与されたアイテムを“低評価アイテム群”と定義している。そして、Item2Vec の学習を行う際、同じユーザに評価され、かつ同じアイテム群に分類されたアイテムの分散表現が類似するように学習を行う。その際、Kuzmin の手法では、高評価アイテム群と低評価アイテム群を同時に Item2Vec で学習し、単一の意味空間を構成している。しかし、ユーザが対象ア

アイテムに対して高評価する理由と、低評価する理由は異なると考えられる。そのため、それらを単一のモデルで表現するのではなく、別々のモデルで表現することで、ユーザの嗜好をより捉えたアイテムの分散表現が可能になると考えられる。

そこで本研究では、各ユーザに対して与えられる高評価アイテム群と低評価アイテム群を、それぞれ独立に学習し、異なる意味空間を構成する。そして、これらを統合する形で評価値予測を行う手法を提案する。また、提案手法を映画評価履歴データに適用し、従来手法と比較することで、提案手法の有効性を検証する。

## 2. 準備 - Item2Vec

自然言語処理の分野において、単語を低次元の意味空間上のベクトルで表現する手法として Word2vec [Mikolov 2013] が知られている。ここで Word2vec をアイテムの分散表現モデルに援用した手法が Item2Vec である。ユーザが過去に購買した全アイテムを購買系列として、あるアイテムは、同じ購買系列中の他のアイテムから予測できると仮定する。そして、注目アイテムベクトルと周辺アイテムベクトルとの内積を大きくするように各アイテムベクトルの学習を行う。また、学習の進行を高速化し、分散表現の精度を高めるために、ネガティブサンプリング [Mikolov 2013] が一般的に用いられている。ネガティブサンプリングでは、全アイテムから確率的にアイテムのサンプリングを行うノイズ分布をあらかじめ定義し、注目アイテムに対してそのアイテム系列とは関係なく獲得されるアイテム (ネガティブアイテム) をこのノイズ分布に従ってサンプリングする。そして、ネガティブアイテムと注目アイテムとの内積を小さくするように各アイテムベクトルを更新する。

いま、アイテム数を  $N$ 、ユーザ数を  $M$  とし、全アイテム集合を  $\mathcal{I} = \{i_n : 1 \leq n \leq N\}$ 、全ユーザ集合を  $\mathcal{U} = \{u_m : 1 \leq m \leq M\}$  とする。また、ネガティブサンプリング数を  $K$  とする。ユーザ  $u_m$  が  $j$  番目に購買したアイテム  $x_{m,j} \in \mathcal{I}$  を表す意味空間上のベクトルを  $\mathbf{v}_{m,j}$  とし、ある 1 つの周辺アイテム  $x_{m,l} (l \neq j) \in \mathcal{I}$  を表すベクトルを  $\mathbf{v}_{m,l}$  とする。また、ネガティブアイテム  $y_{m,j}^k \in \mathcal{I} (1 \leq k \leq K)$  のベクトルを  $\mathbf{s}_{m,j}^k$  で表す。このとき、注目アイテム  $x_{m,j}$  に対する周辺アイテム

連絡先: 後藤 亮介, 早稲田大学院創造理工学研究科経営システム工学専攻, 169-8555, 東京都新宿区大久保 3-4-1 51 号館 15 階 ps wonder@akane.waseda.jp

への損失関数は式 (1) で定義される。

$$Loss(x_{m,j}, x_{m,l}) = \log(\sigma(\mathbf{v}_{m,j}^T \cdot \mathbf{v}_{m,l})) + \sum_{k=1}^K \log(\sigma(-\mathbf{v}_{m,j}^T \cdot \mathbf{s}_{m,j}^k)) \quad (1)$$

ただし、 $\sigma(\cdot)$  はシグモイド関数を表す。ユーザ  $u_m$  の全購買アイテム数を  $e_m$  とすると、学習データ全体における損失項は以下の式 (2) で定義される。

$$Loss_{All} = \sum_{m=1}^M \sum_{j=1}^{e_m} \sum_{l \neq j}^{e_m} Loss(x_{m,j}, x_{m,l}) \quad (2)$$

### 3. 従来手法 - Kuzmin の手法

従来の Item2Vec は各ユーザの購買系列を 1 つの入力データとして学習を行う手法である。仮に、Item2Vec を評価の有無のデータにそのまま適用すると、あるユーザが高評価したアイテムと低評価したアイテムが同じように扱われて学習されてしまうという問題点がある。そこで、Kuzmin の手法では、各ユーザの平均評価値を閾値として、全アイテムを高評価アイテム群と低評価アイテム群に分割する。そして、同一アイテム群を 1 つの購買系列と捉えて Item2Vec の学習を行う。これにより、同じユーザに選択されていて、かつ似た評価がされているアイテム同士が、類似したベクトルとなるように分散表現が学習される。そして、得られたアイテムベクトル同士の類似度を用い、式 (3) によって予測評価値を算出する。

$$\hat{r}_{m,x} = \eta_x + \frac{\sum_{\tilde{x} \in \tilde{I}_m} Sim(\tilde{x}, x) \times (r_{m,\tilde{x}} - \eta_{\tilde{x}})}{\sum_{\tilde{x} \in \tilde{I}_m} |Sim(\tilde{x}, x)|} \quad (3)$$

ここで、 $\hat{r}_{m,x}$  はユーザ  $u_m$  のアイテム  $x \in \mathcal{I}$  に対する予測評価値、 $\eta_x$  は全ユーザのアイテム  $x$  に対する平均評価値、 $Sim(\tilde{x}, x)$  はアイテム  $\tilde{x}$  と  $x$  を表すベクトルのコサイン類似度、 $\tilde{I}_m \in \mathcal{I}$  はユーザ  $u_m$  が評価した中でアイテム  $x$  と類似度が高いアイテム集合とする。そして、最終的に各ユーザに対して予測評価値が上位のアイテムを推薦することで、評価値を考慮した推薦システムを実現している。

## 4. 提案手法

### 4.1 概要

従来の Kuzmin の手法では、評価値の高いアイテムと低いアイテムを同一のモデルで学習するため、得られる意味空間は単一である。しかし、ユーザの評価行動の背後には多様な嗜好が存在している。例えば、1 人のユーザに関して考えた場合も、あるアイテムに対して好きと認識するに至るまでのメカニズムと、嫌いと認識するに至るまでのメカニズムは異なると考えられ、これらを単一のモデルで表現することは困難である可能性がある。例えば、ある映画 a, b は共に「アクションシーンが面白い」という理由で高評価されやすいが、一方、映画 a は「ストーリーがつまらない」、映画 b は「俳優が魅力的ではない」という理由で低評価されやすいというように、高評価では同一の評価基準の 2 つの映画でも、低評価では評価基準が異なる場合がある。この場合、映画 a と b を単一のモデルで表現した際に、高評価される基準で類似度が高い映画と低評価される基準で類似度が高い映画が異なり、学習がうまくできないことが考えられる。

そこで本研究では、高評価アイテム群と低評価アイテム群に関して、それぞれ独立の意味空間を構成する (以下、高評価ベクトル空間、低評価ベクトル空間と呼ぶ)。すなわち、高評価アイテム群のみを学習させ、高評価アイテム群に特化した分散表現を獲得できるモデルと、低評価アイテム群のみを学習させ、低評価アイテム群に特化した分散表現を獲得できるモデルを獲得する。各アイテム群に対して独立に意味空間を構成することで、評価による意味の違いを捉えたベクトルが得られると考えられる。そして、それぞれのモデルから得られた分散表現を統合する形で評価値の予測を行うことで、より高精度な推薦システムの実現を図る。

### 4.2 提案手法のアルゴリズム

提案手法のアルゴリズムを以下に示す。

#### STEP1: アイテム評価系列の分割

各ユーザに対して、ユーザが付与した全評価値の平均を算出する。そして、算出した平均評価値よりも高く評価されているアイテムと低く評価されているアイテムに分割する。

#### STEP2: 意味空間の構成

STEP1 で得られた各アイテム群を学習データとし、Item2Vec を学習する。この際、高評価アイテム群と低評価アイテム群はそれぞれ独立のモデルで学習を行う。

#### STEP3: 2 つの意味空間での予測評価値の算出

各意味空間上のアイテムベクトル同士の類似度を用いて、式 (3) により別々に予測評価値を算出する。

#### STEP4: 予測評価値の算出

式 (3) によって求めた各予測評価値の加重平均により、最終的な予測評価値を式 (4) で算出する。

$$\hat{r}_{m,x} = \alpha \times \hat{r}_{m,x}^{(low)} + (1 - \alpha) \times \hat{r}_{m,x}^{(high)} \quad (4)$$

ここで、 $\hat{r}_{m,x}$  はユーザ  $u_m$  のアイテム  $x$  に対する予測評価値、 $\hat{r}_{m,x}^{(low)}$  は低評価ベクトル空間上で得られる予測評価値、 $\hat{r}_{m,x}^{(high)}$  は高評価ベクトル空間上で計算される予測評価値、 $\alpha$  は評価値の混合比とする。 $\hat{r}_{m,x}^{(low)}$  は、低評価アイテム群の中に存在する評価軸を考慮したベクトル空間から算出された予測評価値、つまり、低評価アイテム群に特化した予測評価値であると言える。同様に、 $\hat{r}_{m,x}^{(high)}$  は、高評価アイテム群に特化した予測評価値であると言える。

## 5. 実験

### 5.1 実験条件

評価値の予測実験には、公開データセット MovieLens [Lens] の映画評価データを用いる。評価値データ数は 100,000 件、ユーザ数 942 人、アイテム数 1,683 本、評価値は 1~5 の 5 段階を実験対象データである。本実験では評価指標として式 (5) MAE (平均絶対誤差) を用い、5 分割交差検証法によりテストデータに対して得られる各予測精度の平均を求める。

$$MAE = \frac{1}{D} \sum_{m=1}^M \sum_{x \in \mathcal{I}} |\hat{r}_{m,x} - r_{m,x}| \delta_{m,x} \quad (5)$$

ここで、 $r_{m,x}$  はユーザ  $u_m$  のアイテム  $x$  に対する真の評価値、 $D$  はテストデータ数である。また、 $\delta_{m,x}$  はインジケータ関数であり、ユーザ  $u_m$  がアイテム  $x$  を評価している場合は 1、それ以外は 0 を示す。

さらに、2つのベクトル空間のバランスを調整するために導入する、高評価ベクトル空間と低評価ベクトル空間での予測評価値の混合比 $\alpha$ は、学習データをさらに設定用学習データと設定用テストデータに分割し、交差検証法により最適化する。具体的には、設定用学習データでItem2Vecを行い2つのベクトル空間を構成して設定用テストデータの評価値を予測する。さらに、混合比 $\alpha$ を0から1まで0.1ずつ変化させた全パターンに対してMAEを求める。そして、交差検証法により分割された5つの設定用テストデータに対して求めたMAEの平均が最も低い $\alpha$ を最適な混合比とする。また、高評価アイテム群と低評価アイテム群の分散表現モデルの違いを確認するために、ベクトル空間が異なることによるアイテム間の類似度の差異についても検証する。

## 5.2 実験結果と考察

図1は、混合比 $\alpha$ を0から1まで変化させた場合の設定用テストデータに対して求めたMAEの推移を表す。

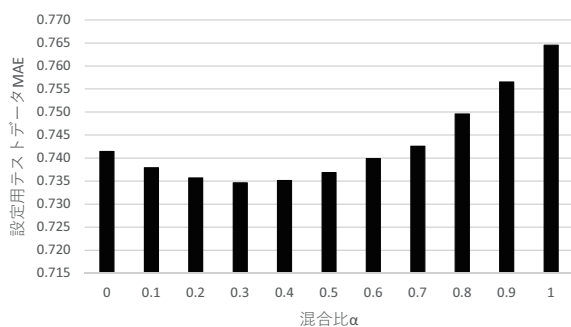


図1: 混合比を変化させた際のMAEの推移

図1より、 $\alpha=0.3$ のときに最もMAEが小さくなった。そこで、最適な混合比 $\alpha$ を0.3と設定した時の評価値予測実験の結果を表1に示す。また、比較手法として従来手法と高評価ベクトル空間、低評価ベクトル空間の片方のみを用いて評価値予測をした結果を示す。

表1: 各手法におけるMAE

手法	MAE
従来手法 (Kuzminの手法)	0.7263
高評価ベクトル空間のみを用いた場合 ( $\alpha=0.0$ )	0.7264
低評価ベクトル空間のみを用いた場合 ( $\alpha=1.0$ )	0.7434
提案手法 ( $\alpha=0.3$ )	<b>0.7214</b>

表1より、提案手法は従来手法と比べて予測精度が向上していることが分かる。高評価ベクトル空間のみを用いた場合と低評価ベクトル空間のみを用いた場合において予測精度が低下したのは、学習データを高評価と低評価に分割する必要があり、モデルの学習に用いるデータが相対的に少なくなってしまうためと考えられる。しかし、これらのモデルを統合して評価値を予測することで、アンサンブル効果により従来法よりも予測精度を向上させることが可能であることが表1からも確認できる。

以降では、高評価アイテム群と低評価アイテム群の分散表現モデルの違いを確認するために、ベクトル空間が異なることによるアイテム間の類似度の差異について、実際のアイテムの組み合わせの例を用いて検証を行う。まず、表2に、高評価ベクトル空間、低評価ベクトル空間で共に類似度が高いアイテムの組み合わせの一例を示す。

表2: 高評価ベクトル空間、低評価ベクトル空間で共に類似度が高いアイテムの組み合わせ

映画タイトル	高評価ベクトル空間	低評価ベクトル空間
スタートレック4 スタートレック6	0.8540	0.9068
フリッパー パーフェクトワールド	0.7697	0.7990
パイオドーム クロウ	0.7651	0.7486
Go FisH カジノ	0.8523	0.8921
スポン スニーカーズ	0.8132	0.8189
StarWars6 キリングフィールド	0.8239	0.8432
めぐり逢えたら 星に願いを	0.8334	0.8336
奴らに深き眠りを コンタクト	0.8665	0.8410

表2より、例えばスタートレック4とスタートレック6は、高評価ベクトル空間と低評価ベクトル空間で、共に高い類似度を示した。すなわち、スタートレック4を高く評価しているユーザは、スタートレック6を同時に高く評価している。また、スタートレック4を低く評価しているユーザも、同じくスタートレック6を低く評価していることがわかる。そのため、2つの作品は同一ユーザに見られやすく、かつ、評価の基準の軸も似ていると言える。

次に、表3に、高評価ベクトル空間、低評価ベクトル空間で共に類似度が低いアイテムの組み合わせの一例を示す。

表3: 高評価ベクトル空間、低評価ベクトル空間で共に類似度が低いアイテムの組み合わせ

映画タイトル	高評価ベクトル空間	低評価ベクトル空間
Sweet Hereafter 地球が静止する日	0.2239	0.2698
クルックリン ゲーム	0.2392	0.1881
シャイニング 評決のとき	0.4183	0.3997
戦火の勇気 タイタニック	0.2604	0.1866
おしゃれキャット Critical Care	0.2969	0.2780
ジョーズ2 ゲース	0.2941	0.2441
クラッシュ ジョージア	0.2291	0.2246
シャイニング 評決のとき	0.4183	0.3997
エマ ベン・ハー	0.2677	0.2924

表3より、例えばSweet Hereafterと地球が静止する日は、高評価ベクトル空間と低評価ベクトル空間で、共に低い類似度を示した。すなわち、Sweet Hereafterを高く評価しているユーザは地球が静止する日を高く評価しないことが分かる。また、Sweet Hereafterを低く評価しているユーザは地球が静止する日を低く評価しないことが分かる。このことから、ユーザが評価する際の基準の軸が2つの作品では全く異なるという

ことが分かる。

そして、表4に、高評価ベクトル空間では類似度が高く、低評価ベクトル空間では類似度が低いアイテムの組み合わせの一例を示す。

表4: 高評価ベクトル空間では類似度が高く、低評価ベクトル空間では類似度が低いアイテムの組み合わせ

映画タイトル	高評価ベクトル空間	低評価ベクトル空間
ニューシネマパラダイス StarWars5	0.7017	0.3456
トランスフォーマー フェノミナン	0.6573	0.3570
セブン ニキータ	0.6215	0.3519
ニューシネマパラダイス ライフ・オブ・ブライアン	0.7012	0.4338
クリムゾン・タイド フrintストーン	0.7212	0.4201
グッドフェローズ ファーザーズ・デイ	0.5746	0.3586
スリング・ブレイド シャイニング	0.6211	0.3525
アナコンダ ディアボロス	0.5769	0.3802

表4より、例えば、ニューシネマパラダイスを高く評価しているユーザは、StarWars5も高く評価している傾向があるが、低評価ではその傾向が見られない。つまり、2つの作品がユーザに高評価される基準は同じ傾向があるが、低評価される基準の軸が作品によって異なることを意味する。

最後に表5に、高評価ベクトル空間では類似度が低く、低評価ベクトル空間では類似度が高いアイテムの組み合わせの一例を示す。

表5: 高評価ベクトル空間では類似度が低く、低評価ベクトル空間では類似度が高いアイテムの組み合わせ

映画タイトル	高評価ベクトル空間	低評価ベクトル空間
Uターン 大脱走	0.1970	0.6002
トップガン バットマンリターンズ	0.4928	0.8012
ガス燈 ワン・ナイト・スタンド	0.4960	0.8007
ピノキオ パッドボーイズ	0.4743	0.7332
クルックリン トゥルー・クライム	0.4726	0.7862
未来は今 ピアノ・レッスン	0.4314	0.7639
ベン・ハー アンフォゲッタブル	0.2749	0.7276
ジュニア 理由	0.3983	0.7845

表5より、例えば、Uターンを低く評価しているユーザは大脱走も低く評価している傾向があるが、高評価ではその傾向が見られない。つまり、2つの作品がユーザに低評価される基準は同じ傾向があるが、高評価される基準の軸が作品によって異なることを意味する。

このように、高評価アイテム群の背後に存在するユーザの嗜好と、低評価アイテム群の背後に存在するユーザの嗜好を独立に学習することで、ユーザの嗜好を詳細に把握することを可能とし、予測精度の向上に寄与していると考えられる。

## 6. まとめと今後の課題

本研究では、よりユーザの嗜好を捉えたアイテムの分散表現を獲得するために、高評価ベクトル空間と低評価ベクトル空間を独立に学習を行う手法を提案した。さらに、2つのベクトル空間のバランスを調整するための混合比を導入することで、各ベクトル空間上から推定される評価値の加重平均で評価値予測を行うモデルを提案した。また、ベンチマークデータを用いた実験により提案手法の有効性を示した。さらに、ベクトル空間毎のアイテムの類似度の組み合わせを4種類示し、提案手法が予測精度の向上に寄与する原因について考察を行った。

今後の課題としては以下の3点が挙げられる。本提案モデルでは、異なるベクトル空間上で予測した評価値を統合するために、混合比 $\alpha$ を全アイテムに対して一律に設定したが、アイテムによって異なる $\alpha$ を決定する方法について検討する余地がある。また、本提案モデルにおいては、公開データセットで実験を行い有意性を検証したが、異なるデータセットに対して提案手法を適用させ、汎化性能を検証する必要がある。最後に、評価値予測だけでなくTopN精度でアイテムに対する推薦精度の測定をすることが挙げられる。

## 参考文献

- [神寫 2003] T.Kamishima, “Nantonac collaborative filtering : recommendation based on order responses,” in *Proc. of The 9th International Conference on Knowledge Discovery and Data Mining*, pp. 583-588, Aug, (2003).
- [Kuzmin 2017] Vitali Kuzmin, “Item2Vec-based Approach to a Recommender System,” *University of Tarutu*, (2017).
- [Barkan 2016] Oren Barkan, Noam Koenigstein, “Item2Vec: Neuralitem embedding for collaborative filtering,” *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing*, (2016).
- [Mikolov 2013] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” *ICLR Workshop*, (2013).
- [Mikolov 2013] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” *Advances in NIPS26*, pp. 3111-3119, (2013).
- [Lens] MovieLens Dataset, <https://grouplens.org/datasets/movielens/>, 2019年1月18日閲覧。