

ユーザーの特性を考慮した音楽推薦アルゴリズム

太刀岡勇気†

† デンソーアイティラボラトリー 〒150-0002 東京都渋谷区渋谷 2-15-1 渋谷クロスタワー 28F
E-mail: †tachioka.yuki@core.d-itlab.co.jp

あらまし 推薦の際にユーザーの特性がバイアスを生むことが知られている。ユーザーの特性とは OCEAN モデルとして知られる、経験への開放性・誠実性・外向性・協調性・神経症傾向がある。変分オートエンコーダ (VAE) を用いた推薦アルゴリズムが近年高い性能を示している。VAE の推薦アルゴリズムに、ユーザーの特性を考慮に入れることで、さらに性能を向上させられる可能性がある。そこで変分オートエンコーダを条件付き変分オートエンコーダに変更し、ユーザーの特性をラベルとして入力する方法を提案する。twitter の履歴から推定されるユーザーの特性ラベルが付与された音楽推薦のデータセットを用いて、提案法の有効性を明らかにした。

キーワード 推薦モデル, ユーザー特性, OCEAN モデル, 条件付き変分オートエンコーダ

1 はじめに

商品の推薦の際には、ユーザーの嗜好に合わせることで満足度向上に重要であるが、その際にユーザーの特性がバイアスを生むことが知られている。ユーザーの特性としては様々なものが考えられるが、一例として広く使われているものとしては表 1 に示す OCEAN(BigFive) モデルの 5 因子がある [3]。実際、文献 [2] では、音楽推薦タスクにおいて、この 5 因子のポジティブ・ネガティブが推薦手法の性能に有意な差を与えることを示している。

近年推薦アルゴリズムには、深層学習をベースにしたものが用いられている。中でもディノイズングオートエンコーダ (DAE) を用いる方法 [6] がよく用いられる。しかしながら、推薦タスクはデータが非常にスパースであるため、DAE は過学習しやすい傾向にある。そのため、隠れ変数に特定の分布を仮定する変分オートエンコーダ (Variational autoencoder; VAE) を用いた方法が有効とされる。実際、多項分布 (Multinomial distribution) を仮定した MultVAE が、従来の DAE よりも高い性能を示すことが知られている [1]。

ただし、5 因子が推薦結果に影響を及ぼすのであれば、この分布も 5 因子に応じてユーザーごとに異なる可能性がある。条件付き変分オートエンコーダを推薦タスクに導入した従来法の例としては [7] があるが、ユーザーの特性を入力とした従来研究は見られない。そこで、ユーザーの特性を考慮した推薦を行う方法が有効であるか検討を行う。具体的には、MultVAE においてユーザーの特性を考慮に入れるために、変分オートエンコーダを

条件付き変分オートエンコーダ (Conditional VAE; CVAE) [5] に変更し、ユーザーの特性をラベルとして入力する方法を提案する。さらに前述のユーザーの twitter の履歴から推定されるユーザーの特性ラベル (OCEAN モデルで表される BigFive 因子) が付与された音楽の楽曲推薦のタスクを題材として、提案法の有効性を検証する。

2 MultDAE/MultVAE

この節では、本報でベースラインとして採用している MultVAE [1] について触れる。図 1 に DAE/VAE の構造を示す。DAE は入力 \mathbf{x} に対してエンコーダ ϕ により隠れ変数 \mathbf{z} に変換し、それをデコーダ θ により元の入力を再現した出力 \mathbf{x}' を得るものである。 \mathbf{x} はアイテム数の次元を持つ疎ベクトルであり、ユーザーごとに興味のあるアイテムの部分のみ 1 となっているベクトルである。学習に用いるユーザーは \mathbf{x} のすべてが埋められている。テストに用いるユーザーは、 \mathbf{x} の一部を隠してオートエンコーダに入力し、推薦リスト \mathbf{x}' を生成させ、隠した部分の推薦結果により性能を評価する。変換の頑健性を向上させるため、オートエンコーダを用いる方法にはノイズ ϵ を加えることが一般的である。

これに対して VAE は、隠れ変数 \mathbf{z} そのものをエンコーダで推定するのではなく、平均 μ と分散 σ^2 のみを推定し、隠れ変数は仮定した生成モデルに従うものとして生成される点が異なる。MultVAE [1] では、隠れ変数の分布には、多項分布を仮定する。VAE ではデータの尤度 $p(\mathbf{x})$ を最大化することが目的であるが、これを直接行うことはできないため、近似事後分布 q_ϕ を用いて ELBO (evidence lower bound) の最大化を行う。結果として、損失関数 L は、負の再構成誤差評価項と KL 擬距離による正則化項の重みづけとなる。

$$\mathcal{L}(\mathbf{x}; \theta, \phi) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta KL(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) \quad (1)$$

表 1 OCEAN(BigFive) モデルの 5 因子.

経験への開放性	Openness to experience (ope)
誠実性	Conscientiousness (con)
外向性	Extroversion (ext)
協調性	Agreeableness (agr)
神経症傾向	Neuroticism (neu)

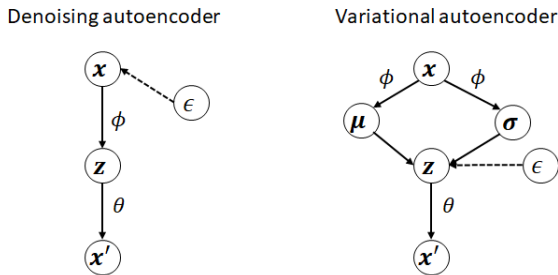


図1 推薦タスクに用いられるオートエンコーダ (MultDAE) と変分オートエンコーダ (MultVAE).

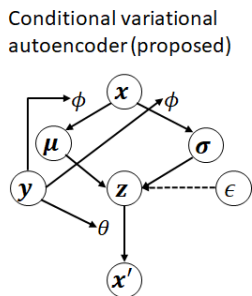


図2 提案の条件付き変分オートエンコーダ (MultCVAE).

3 提案法 (MultCVAE)

エンコーダ ϕ とデコーダ θ の学習時には、学習用に用意したユーザのデータすべてを用いる。よって、生成される推薦リスト \mathbf{x}' は学習データのすべてのユーザの嗜好を反映したものとなる。このため、全く自分と特性の違うユーザの好みも含まれているので、推薦の精度が下がる可能性がある。

そこで提案法では、ユーザ特性のラベル \mathbf{y} を用いて¹、エンコーダ ϕ とデコーダ θ を条件づける方法を提案する。これにより自分と近い特性を持つユーザの嗜好を重視して、推薦リスト \mathbf{x}' を生成することができる。図2に前節の MultVAE を拡張した提案法の構造を示す。条件付き VAE (CVAE) の実装は [5] および公開されているコード²を参考にし、エンコーダには \mathbf{x} を全結合層で低次元に変換したのちにラベル \mathbf{y} を結合して入力した。デコーダには \mathbf{z} と \mathbf{y} を結合して与えた。

4 実験

4.1 実験条件

ユーザの特性ラベルが付与された音楽推薦のデータセット [2]³ を用いて評価を行った。15,753 トラックに対して、395,056 のリスニングイベントがある。 \mathbf{x} はユーザのこれまで聞いた曲目リストで、 $\mathbf{x} \in \{0, 1\}^{15753}$ である。18,310 人のユーザに、ユーザ特性の値がつけられている。ユーザ特性値として、BigFive の各因子の強さを実数値で表した 5 次元の実数値ベクトルが付

与されている。ユーザの特性はリツイートを除く最新 1000 ツイートから IBM personality Insight API に基づき推定されている。学習にはこのうちの 8 割を用い、2 割をテストデータとした。テストデータは [2] での評価に倣い、全ユーザと特性がポジティブなユーザのみでの評価の 2 通り用意した⁴。

MultDAE/VAE とともにエンコーダでは全結合層を 2 層用い、 \mathbf{x} を 600 次元のベクトルに変換したのちに 200 次元の \mathbf{z} を得る。エンコーダとデコーダは対称な構造である。提案の CVAE ではエンコーダの 600 次元のベクトルにラベル \mathbf{y} を結合することでエンコーダを条件付け、また \mathbf{z} にラベル \mathbf{y} を結合することでデコーダを条件付けした。ハイパーパラメータは [1] を参考に設定した。50 エポック学習を行い、検証データの $NDCG@50$ が最良のモデルでテストデータを評価した。比較として RecVAE [4] の性能も評価した。実装は公開されているコード⁵をベースとした。提案法 (MultCVAE) のラベル \mathbf{y} には、表 2 に示す 3 種類を用意した。バイナリラベル作成のために、BigFive の特性ごとに中央値を算出し、それ以上のユーザをポジティブ、未満のユーザをネガティブとした。

評価指標には式 (2) で示される $Recall@K$ と式 (4) で示される $NDCG@K$ を用いた。 $rel(i)$ は指示関数であり、アイテム i をユーザが好んだ場合に 1、それ以外の場合に 0 となる関数である。ユーザにテストセットで提示した N アイテムのうち K の推薦リストのうち rel が 1 であった割合が Recall である。

$$Recall@K = \frac{1}{\min(K, N)} \sum_{i=1}^K rel(i) \quad (2)$$

NDCG は、システムがユーザに提示したアイテムが実際にユーザに好まれる (再生回数が多い) ものとどの程度近いかを表す。DCG は、システムがユーザに提示したアイテムを推薦リストの位置によって割り引いている。

$$DCG@K = \sum_{i=1}^K \frac{rel(i)}{\log_2(i+1)} \quad (3)$$

そして IDCG は、実際のユーザの再生回数が多いものから降順にソートした場合の DCG である。NDCG は、DCG を IDCG で除して得られる。

$$NDCG@K = \frac{DCG@K}{IDCG@K} \quad (4)$$

$NDCG@100$ は推薦リストの上位 100 位までが自分の好みの順番とどの程度一致しているかを示す。

表 2 提案法 (MultCVAE) のラベル \mathbf{y} .

ポジティブネガティブのバイナリラベル (pos-neg)
5 因子中の最大値をとる因子のインデックスの one-hot ベクトル (Big5)
BigFive の実数のベクトルそのもの (Big5(vec))

1: 以下の実験ではユーザの特性として OCEAN (BigFive) モデルを採用する。

2: <https://github.com/chendaichao/VAE-pytorch>

3: https://github.com/CPJKU/pers_bias

表 3 全ユーザで評価した NDCG.

	RecVAE	MultDAE	MultVAE	MultCVAE		
				pos-neg	Big5	Big5(vec)
NDCG@50						
ope	0.0554	0.0611	0.0610	0.0589	0.0622	0.0577
con	0.0566	0.0634	0.0648	0.0645	0.0673	0.0679
ext	0.0610	0.0651	0.0630	0.0643	0.0660	0.0616
agr	0.0543	0.0599	0.0599	0.0602	0.0626	0.0611
neu	0.0563	0.0624	0.0601	0.0631	0.0599	0.0591
avg	0.0567	0.0624	0.0618	0.0622	0.0636	0.0615
NDCG@100						
ope	0.0673	0.0742	0.0729	0.0709	0.0740	0.0686
con	0.0671	0.0768	0.0767	0.0770	0.0796	0.0822
ext	0.0720	0.0776	0.0750	0.0764	0.0782	0.0736
agr	0.0646	0.0734	0.0722	0.0733	0.0747	0.0727
neu	0.0663	0.0751	0.0727	0.0746	0.0729	0.0704
avg	0.0675	0.0754	0.0739	0.0744	0.0759	0.0735

表 4 全ユーザで評価した Recall.

	RecVAE	MultDAE	MultVAE	MultCVAE		
				pos-neg	Big5	Big5(vec)
Recall@20						
ope	0.0696	0.0797	0.0826	0.0784	0.0770	0.0777
con	0.0695	0.0772	0.0802	0.0811	0.0834	0.0848
ext	0.0760	0.0852	0.0810	0.0796	0.0816	0.0801
agr	0.0707	0.0794	0.0772	0.0776	0.0787	0.0810
neu	0.0702	0.0802	0.0773	0.0761	0.0745	0.0758
avg	0.0712	0.0803	0.0797	0.0786	0.0790	0.0799
Recall@50						
ope	0.1086	0.1239	0.1279	0.1235	0.1268	0.1195
con	0.1109	0.1319	0.1269	0.1296	0.1359	0.1326
ext	0.1183	0.1341	0.1299	0.1271	0.1326	0.1211
agr	0.1086	0.1306	0.1308	0.1283	0.1305	0.1233
neu	0.1173	0.1300	0.1220	0.1321	0.1225	0.1218
avg	0.1127	0.1301	0.1275	0.1281	0.1297	0.1237

4.2 結果と考察 (全ユーザで評価)

全ユーザで評価した $NDCG@K$ を表 3 に示す. すべての条件で MultCVAE が MultVAE の性能を上回った. これより条件づけによる有効性が示された. 2 パターンのみ MultDAE が最良の性能を示したが, それ以外の条件では提案法の性能が最良であった. 平均的には BigFive の特性値の実数値のベクトルを直接入力 (Big5(vec)) するよりも, one-hot ベクトルをラベルに用いたもの (Big5) が最も性能が高かった.

同じく Recall を表 4 に示す. Recall の観点では ope のみ従来法に性能が劣ったが, それ以外の条件では性能を改善した. ただこの場合は NDCG ほどの改善は見られなかった. これはモデルの最適化を, $NDCG@50$ の観点で行っているためと考えられる.

4: [2] には特性がネガティブなユーザのみの評価も含まれるが, それは提案法の意図する似た特性のユーザの推定結果を積極的に利用するという使われ方ではないため, この 2 通りとした.

5: <https://github.com/ilya-shenbin/RecVAE>

表 5 特性がポジティブなユーザのみで評価した NDCG.

	RecVAE	MultDAE	MultVAE	MultCVAE		
				pos-neg	Big5	Big5(vec)
NDCG@50						
ope	0.0490	0.0538	0.0540	0.0527	0.0574	0.0510
con	0.0538	0.0602	0.0652	0.0649	0.0647	0.0702
ext	0.0607	0.0630	0.0639	0.0598	0.0641	0.0656
agr	0.0570	0.0641	0.0643	0.0638	0.0669	0.0656
neu	0.0673	0.0706	0.0697	0.0717	0.0706	0.0695
avg	0.0576	0.0623	0.0634	0.0626	0.0647	0.0644
NDCG@100						
ope	0.0602	0.0657	0.0650	0.0635	0.0690	0.0609
con	0.0644	0.0734	0.0764	0.0774	0.0771	0.0840
ext	0.0712	0.0744	0.0743	0.0720	0.0763	0.0765
agr	0.0673	0.0769	0.0763	0.0770	0.0796	0.0783
neu	0.0768	0.0843	0.0820	0.0836	0.0838	0.0814
avg	0.0680	0.0749	0.0748	0.0747	0.0772	0.0762

表 6 特性がポジティブなユーザのみで評価した Recall.

	RecVAE	MultDAE	MultVAE	MultCVAE		
				pos-neg	Big5	Big5(vec)
Recall@20						
ope	0.0588	0.0705	0.0713	0.0701	0.0703	0.0714
con	0.0659	0.0718	0.0776	0.0810	0.0835	0.0887
ext	0.0755	0.0860	0.0773	0.0774	0.0826	0.0868
agr	0.0770	0.0870	0.0832	0.0874	0.0861	0.0857
neu	0.0844	0.0898	0.0892	0.0919	0.0866	0.0896
avg	0.0723	0.0810	0.0797	0.0816	0.0818	0.0844
Recall@50						
ope	0.0916	0.1055	0.1107	0.1092	0.1156	0.1006
con	0.1031	0.1245	0.1247	0.1258	0.1248	0.1317
ext	0.1174	0.1299	0.1271	0.1210	0.1249	0.1239
agr	0.1152	0.1379	0.1359	0.1381	0.1298	0.1286
neu	0.1316	0.1402	0.1346	0.1441	0.1370	0.1362
avg	0.1118	0.1276	0.1266	0.1276	0.1264	0.1242

4.3 結果と考察 (特性がポジティブなユーザのみで評価)

次に学習されたモデルは同じまま, 特性ごとにポジティブな (中央値よりも大きな値を取ったおおむね半数の) ユーザのみで評価した. この場合の評価した NDCG を表 5 に示す. こちらは提案法の性能向上が顕著であり, 1 ケースを除いて提案法の性能が最良であった. 傾向は同じで one-hot ベクトルをラベルとしたもの (Big5) が最も性能が高かった.

同じく Recall を表 6 に示す. こちらも同傾向であったが, 表 4 の結果よりも提案法による性能改善が多くみられる.

5 まとめと今後の課題

ユーザの特性を推薦に反映するために, VAE をラベルで条件づける方法を提案した. ユーザの特性として OCEAN モデルから得られる BigFive の 5 因子を採用し, ラベルとして, ポジティブ・ネガティブのバイナリ値, BigFive の最大値のインデックス (one-hot), BigFive の実数値ベクトル (連続値) の 3 種類を比較した. 音楽データを用いた実験により, BigFive の

最大値のインデックスが、NDCG の観点ではもっとも性能を改善することがわかった。これにより、ユーザの特性を考慮した推薦を行う提案法が、ユーザの特性を考慮しない従来法よりも推薦の精度を向上させることがわかった。今後は他のタスクでも検証する予定である。

文 献

- [1] Liang, D., Krishnan, R. G., Hoffman, M. D. and Jebara, T.: Variational Autoencoders for Collaborative Filtering, *Proceedings of the 2018 World Wide Web Conference, WWW '18*, Republic and Canton of Geneva, CHE, International World Wide Web Conferences Steering Committee, pp. 689–698 (online), <https://doi.org/10.1145/3178876.3186150> (2018).
- [2] Melchiorre, A. B., Zangerle, E. and Schedl, M.: Personality Bias of Music Recommendation Algorithms, *Fourteenth ACM Conference on Recommender Systems, RecSys '20*, New York, NY, USA, Association for Computing Machinery, pp. 533–538 (online), <https://doi.org/10.1145/3383313.3412223> (2020).
- [3] Rothmann, S. and Coetzer, E. P.: The big five personality dimensions and job performance, *SA Journal of Industrial Psychology*, Vol. 29, No. 1 (online), <https://doi.org/10.4102/sajip.v29i1.88> (2003).
- [4] Shenbin, I., Alekseev, A., Tutubalina, E., Malykh, V. and Nikolenko, S. I.: RecVAE: A New Variational Autoencoder for Top-N Recommendations with Implicit Feedback, *Proceedings of the 13th International Conference on Web Search and Data Mining*, New York, NY, USA, Association for Computing Machinery, pp. 528–536 (online), <https://doi.org/10.1145/3336191.3371831> (2020).
- [5] Sohn, K., Yan, X. and Lee, H.: Learning Structured Output Representation Using Deep Conditional Generative Models, *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2, NIPS'15*, Cambridge, MA, USA, MIT Press, p. 3483–3491 (2015).
- [6] Wu, Y., DuBois, C., Zheng, A. X. and Ester, M.: Collaborative Denoising Auto-Encoders for Top-N Recommender Systems, *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, WSDM '16*, New York, NY, USA, Association for Computing Machinery, pp. 153–162 (online), <https://doi.org/10.1145/2835776.2835837> (2016).
- [7] 保住 純, 岩澤有祐, 松尾 豊: 行動時刻を考慮した条件付き変分オートエンコーダによる推薦システム, *人工知能学会論文誌*, Vol. 36, No. 3C, pp. 1–10 (2021).

6 1 ページ目に関して

1 ページ目上部には、タイトル、発表者氏名、所属、住所、メールアドレス、キーワードの和文と英文及びあらまし (300 字程度) を、それぞれ記述してください。なお、和文論文については英文タイトル、アブストラクト等は削除して頂いて構いません。下記のコマンドで講演番号を挿入して下さい。

```
\papernumber{DEIM Forum 2023 XX-Y}
```

XX はセッション番号 (例: 1A, 3B), Y はセッション内での発表順 (1, 2, ...) です。番号についてはプログラムをご覧ください。なお、プログラム決定前の初回投稿時には XX-Y の部分の記入は不要です。

7 原稿提出枚数

所定のページ数 (4-8 ページ) を厳守してください。Ph.D セッション投稿者は 8 ページを推奨します。

8 原稿の書き方

原稿のスタイルは、A4 サイズで、9 ポイントのフォントを使用し、2 段組み、シングルスペースとして下さい。