

# Zero-shot 推薦における類似ユーザ情報活用の有効性検証

白見太佑 正木晶子 村松沙那恵 坂本啓 江田毅晴  
NTT 株式会社 ソフトウェアイノベーションセンタ  
taisuke.usumi@ntt.com

## 概要

近年, 大規模言語モデルを用いた推薦システムの研究が活発に行われており, 特にパラメータ更新を伴わない Zero-shot 推薦が注目されている. しかし, 既存の Zero-shot 推薦手法の多くは, 推薦対象のユーザ情報のみを用いており, 他のユーザの情報を十分に活用できていない. 本研究では, ユーザプロフィールの構築, 類似ユーザ情報の検索, 推薦タスクの3ステップからなる Zero-shot 推薦手法を用い, 従来手法との比較により, 他のユーザ情報活用の効果を検証する. 実験の結果, 推薦対象ユーザと嗜好が類似したユーザの情報を活用することが推薦精度を向上させること, および類似ユーザ情報の選択手法としてベクトル検索が最も効果的であることを示した.

## 1 はじめに

大規模言語モデル (Large Language Model; LLM) は自然言語処理をはじめとする多様なタスクにおいて高い性能を示しており, その応用は推薦システムにおいても盛んに研究されている. 近年では, LLM の高い言語処理能力を活用し, モデルのパラメータ更新をせずに Zero-shot で推薦タスクを行う研究が注目を集めている [1, 2, 3, 4]. Zero-shot 推薦は, 特定のドメインやタスクへの依存を抑え, より汎用的な推薦を可能にする.

Zero-shot 推薦においては, ユーザの行動履歴 (アイテムの購入履歴や閲覧履歴など) をプロンプトとして与え, LLM により次の行動を予測する Sequential Recommendation に関する研究が進められてきた [1, 2]. 近年では, 行動履歴に加えて, レビューから抽出されたユーザのアイテムに関する嗜好情報を自然言語で表現したユーザプロフィールを活用する手法 [3] が提案されており, 行動履歴のみからは捉えきれない詳細な嗜好情報を取り入れることで, より高精度な推薦を実現している.

しかし, Zero-shot 推薦は, 推薦対象のユーザ情報

のみを用いて推薦を行うため, 推薦タスクにおいて重要とされる他のユーザの行動履歴を活用できず, 十分な推薦性能が得られない可能性がある. 従来の fine-tuning に基づく推薦手法 [5, 6, 7] では, 多数のユーザの行動履歴から学習された潜在的な知識をモデル内部に保持しており, これらの知識を活用することで対象ユーザへの推薦精度を向上できるが, 多くの Zero-shot 推薦手法ではこの情報を活用していない.

そこで本研究では, 他のユーザの行動履歴が Zero-shot 推薦の性能に与える影響を検証する. 具体的には, ユーザプロフィールに記載された嗜好情報を用いて推薦対象のユーザと嗜好が類似したユーザを検索し, 検索によって得られた類似ユーザ情報をプロンプトに組み込むことで推薦を行う. 類似ユーザ情報の検索手法として, (1) ユーザプロフィールの特徴ベクトル間の距離に基づく検索手法, (2) 候補となる類似ユーザ情報から LLM により有用な類似ユーザ情報を選択する手法, (3) 類似ユーザ情報を用いた推薦結果のフィードバックに基づき, 類似ユーザ情報の選択を逐次的に最適化する手法を検証する.

## 2 関連研究

### 2.1 Sequential Recommendation

Sequential Recommendation とは, ユーザの行動履歴を時系列順に並べ, 次にインタラクションがあるアイテムを予測するタスクである. 従来はユーザの行動履歴を用いてモデルを fine-tuning する手法 [5, 6, 7] が主流であったが, 近年では LLM の高度な言語処理能力を活用し, Zero-shot で推薦を行うアプローチが盛んに検討されている [1, 2, 3]. LLMRank [1] は, ユーザの行動履歴のみをプロンプトとして与え, LLM により Zero-shot で次にインタラクションがあるアイテムの予測を実現している. これに対し, PURE [3] は, レビューから抽出したユーザの嗜好情報および

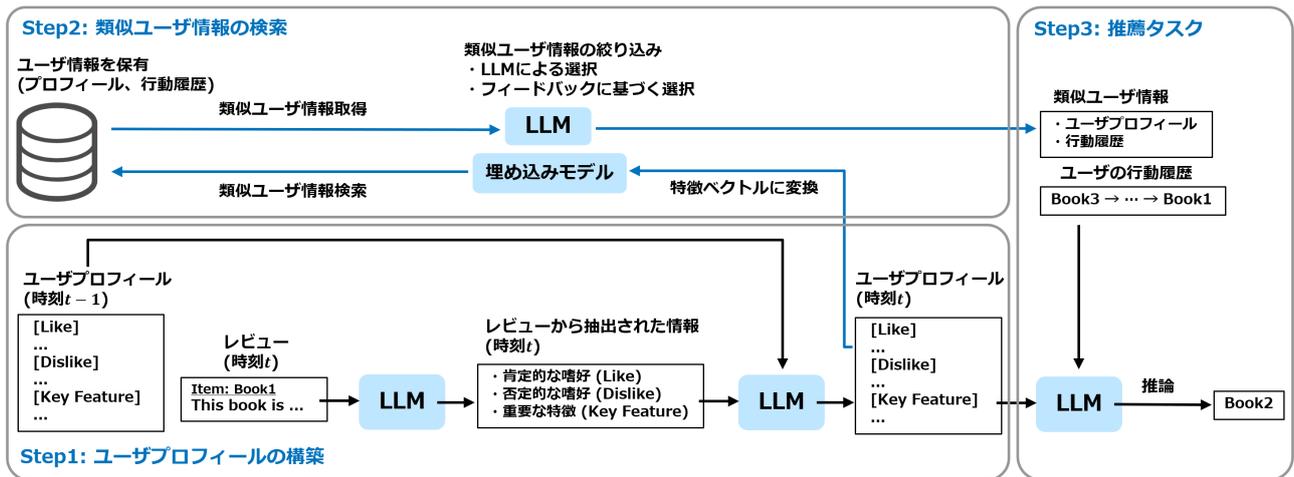


図 1 類似ユーザ情報を用いた推薦手法の概要. 本手法は, (1) ユーザプロフィールの構築, (2) 類似ユーザ情報の検索 (3) 推薦タスクの 3 つのステップから構成される.

Key Feature を活用し, それらをユーザの行動履歴と組み合わせることで, 推薦性能の向上を図っている.

しかし, これらの研究はいずれも対象のユーザ情報のみを推薦タスクに用いており, 従来の fine-tuning を伴う推薦モデルが活用してきた, 他のユーザの情報を推薦過程に取り込むことができていない.

## 2.2 類似ユーザ情報を活用した推薦

LLM を用いた Sequential Recommendation において, 事前に類似ユーザを検索し, その情報を活用して推薦を行うアプローチも提案されている [8, 9]. AdaptRec [8] は, ユーザがインタラクションしたアイテム名のシーケンスの特徴量の類似度に基づいて候補となる類似ユーザを検索し, その後, LLM を用いて最終的な類似ユーザを決定する二段階の類似ユーザ検索手法を提案した. DRDT [9] は, 推薦対象ユーザが直前にインタラクションしたアイテムに着目し, そのアイテムとインタラクションのある他のユーザを類似ユーザとして扱い, 推薦に活用している.

しかし, AdaptRec は fine-tuning を前提としている点で Zero-shot 推薦とは設定が異なる. DRDT においても, 他のユーザ情報は推薦対象ユーザの嗜好分析のために使用しており, Zero-shot 推薦における他のユーザ情報活用の効果については, 十分な検討がなされていない. さらに, いずれの手法においても類似ユーザ検索にアイテム名を用いているため, アイテム間の内容的な類似性を十分に考慮することができず, 結果として嗜好が近いユーザの特定が困難となる可能性がある. これに対し, 本研究では, ユーザの

嗜好を自然言語で表現したユーザプロフィール間の類似度に基づいて, 類似ユーザ情報を検索する.

## 3 手法

図 1 に手法の全体像を示す. 本手法はユーザプロフィールの構築, 類似ユーザ情報の検索, 推薦タスクの 3 つのステップからなる.

### 3.1 タスクの定式化

Sequential Recommendation のタスクを定式化する. まず, ユーザ  $u$  に対してのデータセットとして  $\mathcal{D}_u = \{\mathbf{R}_u, \mathbf{I}_u\}$  を考える. ここで,  $\mathbf{R}_u = \{r_u^1, \dots, r_u^n\}$  はユーザレビューを時系列順に並べたものであり,  $\mathbf{I}_u = \{i_u^1, \dots, i_u^n\}$  はユーザとインタラクションのあるアイテムである. また,  $n$  は行動履歴数である. Sequential Recommendation のタスクは推薦対象のユーザと次にインタラクションがあるアイテム  $i_u^{n+1}$  を各ユーザの候補アイテムリスト  $C_u = \{i_u^j\}_{j=1}^m$  から予測することである.

### 3.2 ユーザプロフィールの構築

ユーザの嗜好を自然言語で表現したユーザプロフィールに基づいて類似ユーザ情報を検索するため, まず PURE [3] の手法を用いて各ユーザのプロフィールを構築する. 具体的には, 時刻  $t$  におけるユーザ  $u$  のレビューから, アイテムに対する肯定的嗜好  $\hat{r}_u^t$ , 否定的嗜好  $\hat{d}_u^t$ , ならびに重要な特徴を示す Key Feature  $\hat{f}_u^t$  を LLM を用いて抽出する. 続いて, 得られた嗜好情報  $(\hat{r}_u^t, \hat{d}_u^t, \hat{f}_u^t)$  を用いて, 時刻  $t-1$  のユーザプロフィール  $\mathbf{P}_u^{t-1} = \{r_u^{t-1}, d_u^{t-1}, f_u^{t-1}\}$  を LLM

により更新することで、時刻  $t$  におけるユーザプロフィール  $\mathbf{P}_u^t$  を構築する。レビューからの嗜好情報および Key Feature の抽出、ならびにユーザプロフィールの構築に用いたプロンプトを付録 A に示す。

### 3.3 類似ユーザ情報の検索

前節で構築したユーザプロフィールを用いて類似ユーザ情報の検索を行う。具体的には、まず全ユーザの各時刻におけるプロフィールを埋め込みモデルにより特徴ベクトルへ変換し、データベースに格納する。そして、推薦時には対象ユーザの最新のプロフィールの特徴ベクトルを用いて嗜好の類似したユーザを検索し、得られた類似ユーザ情報を推薦に利用する。本研究では、類似ユーザ情報の検索手法として、以下のアプローチを検討する。

**ベクトル検索による選択** 推薦対象ユーザの最新のプロフィールの特徴ベクトルとの類似度を用いて上位  $k$  個の類似ユーザプロフィールを取得する。検索対象は、推薦対象ユーザ以外の全ユーザ・全時刻におけるユーザプロフィールである。そのため、検索によって得られる類似ユーザ情報は、特定の時刻におけるユーザ情報である。以下では、これを類似ユーザ情報  $\mathbf{S}_u^k = \{\mathbf{P}_u^t, \mathbf{I}_u^t\}$  と表記し、得られた  $k$  個の類似ユーザ情報の集合を  $\mathbf{U}_u^{\text{sim}} = \{\mathbf{S}_{u_1}^k, \dots, \mathbf{S}_{u_k}^k\}$  とする。

**LLM による選択** ベクトル検索により得られた  $N$  個の類似ユーザ情報の候補から、AdaptRec [8] に倣って LLM により推薦タスクに有用な  $k$  個の類似ユーザ情報  $\mathbf{U}_u^{\text{sim}}$  を選択する。

**フィードバックに基づく選択** LLM によるプロンプト最適化手法である OPRO [10] を応用し、類似ユーザ情報の選択の最適化を行う。具体的には、まずベクトル検索で得られた  $N$  個の類似ユーザ情報から、LLM によって推薦タスクに有用な  $k$  個の類似ユーザ情報  $\mathbf{U}_u^{\text{sim}}$  を選択する。次に、選択された類似ユーザ情報を用いて推薦タスクを実行する。そして、推薦結果が正解であるか否かのフィードバックに基づき、推薦精度が向上するよう類似ユーザ情報の再選択を行うことで、類似ユーザ情報の選択を逐次的に最適化する。なお、最適化には推薦対象ユーザの最新の一つ前のアイテムを正解データとして用いる。

### 3.4 推薦タスク

Zero-shot 推薦において、LLM の入力として最新のユーザプロフィール  $\mathbf{P}_u$ 、行動履歴  $\mathbf{I}_u$ 、類似ユーザ情報  $\mathbf{U}_u^{\text{sim}}$  を用いる。類似ユーザ情報として、類似ユー

ザの行動履歴のみを用いる場合と、行動履歴に加えてプロフィールも用いる場合について評価を行う。推薦タスクに使用したプロンプトの詳細は付録 A に示す。

## 4 実験

### 4.1 データセット

実験には Amazon Reviews [11] の Books と Movies and TV の 2 種類のデータセットを用いる。各データセットには、ユーザ ID、アイテム ID、ユーザレビュー、およびタイムスタンプ情報が含まれている。前処理として、アイテムとのインタラクション数が 21 件未満のユーザを除外した。

### 4.2 実験設定

各データセットから 1032 ユーザを抽出し、そのうち 200 ユーザを推薦対象のユーザ、残りのユーザを類似ユーザの候補として用いる。各ユーザについて直近 21 件のデータを取得し、推薦対象のユーザに関しては最新の 1 件を正解データとし、残りの 20 件をユーザの行動履歴として用いる。候補リストに含まれるアイテム数は 20 件とし、そのうちの 1 件を正解データとする。また、予測に使用する類似ユーザ情報数は  $k=3$  と設定する。各手法について実験を 3 回実施し、その平均値を最終的な評価結果として報告する。

推薦タスクとプロフィール構築の LLM として gpt-4o-mini を使用し、プロフィールを特徴ベクトルに変換する埋め込みモデルとして text-embedding-3-small を用いる。

### 4.3 ベースライン

ベースラインとして以下の類似ユーザ情報を用いない手法を採用する。

**LLMRank [1]**：ユーザの行動履歴のみをプロンプトに入力し、次にインタラクションが生じるアイテムを予測する手法。

**PURE [3]**：ユーザの行動履歴に加えてユーザプロフィールを入力として用い、ユーザの次のインタラクションアイテムを予測する手法。

さらに、類似ユーザ情報の選択手法に関するベースラインとして、以下の手法を用いる。

**ランダム選択** ユーザ情報の集合からランダムに  $k$  個のユーザ情報を取得する。

表1 Zero-shot 推薦における各手法の性能比較.

類似ユーザ情報	Method	Books				Movies and TV			
		N@1	N@5	N@10	N@20	N@1	N@5	N@10	N@20
無し	LLMRank	0.250	0.412	0.475	0.534	0.272	0.506	0.560	0.592
	PURE	0.262	0.444	0.502	0.552	0.292	0.514	0.565	0.598
有り	ランダム選択	0.257	0.434	0.498	0.551	0.302	0.497	0.551	0.589
	直近のアイテムによる選択	0.302	0.474	0.535	0.581	0.320	0.515	0.565	0.602
	ベクトル検索による選択	<b>0.310</b>	<b>0.481</b>	<b>0.545</b>	<b>0.586</b>	<b>0.327</b>	0.515	<b>0.573</b>	<b>0.606</b>
	LLM による選択	0.307	0.478	0.536	0.585	0.322	<b>0.519</b>	0.571	<b>0.606</b>
	フィードバックに基づく選択	0.275	0.454	0.519	0.565	0.305	0.501	0.562	0.595

表2 類似ユーザ情報におけるユーザプロフィール利用の有無による性能比較.

類似ユーザ情報	Books			
	N@1	N@5	N@10	N@20
行動履歴のみ	<b>0.310</b>	<b>0.481</b>	<b>0.545</b>	<b>0.586</b>
行動履歴 + ユーザプロフィール	0.297	0.463	0.533	0.576

表3 類似ユーザ情報数による性能比較.

類似ユーザ情報数	Books			
	N@1	N@5	N@10	N@20
$k = 1$	0.273	0.453	0.510	0.564
$k = 2$	0.273	0.464	0.518	0.569
$k = 3$	<b>0.310</b>	0.481	<b>0.545</b>	<b>0.586</b>
$k = 4$	0.305	<b>0.483</b>	0.544	0.585
$k = 5$	0.292	0.475	0.533	0.580

**直近のアイテムによる選択** DRDT [9] に倣い、直近のインタラクションアイテムが推薦対象ユーザと一致するユーザ情報を最大  $k$  個取得する.

#### 4.4 評価指標

評価指標には、推薦システムの評価に広く用いられている NDCG@K (N@K) を採用する.

## 5 結果

### 5.1 類似ユーザ情報活用の有効性

類似ユーザ情報として行動履歴のみを用いた結果を表 1 に示す. いずれのデータセットにおいても、類似ユーザ情報を活用することで推薦性能の向上が確認され、なかでもベクトル検索に基づく選択手法が最も高い性能を示した. 一方で、ランダムに類似ユーザ情報を選択した場合には性能劣化が見られた. このことは、類似ユーザ情報の適切な選択が性能向上

に不可欠であることを示している.

### 5.2 類似ユーザ情報の種類による性能比較

類似ユーザ情報として行動履歴のみを使用する場合と、行動履歴とユーザプロフィールを用いる場合の推薦性能を比較する. Amazon Reviews の Books データセットを用い、ベクトル検索に基づいて類似ユーザ情報を選択する手法による比較結果を表 2 に示す. 結果として、類似ユーザ情報としてユーザプロフィールを用いた場合に推薦性能の低下が確認された. これは、類似ユーザのプロフィールの追加によりプロンプトが冗長化し、推薦タスクにおいて最も重要である推薦対象ユーザの情報への注目度が低下したことが一因であると考えられる.

### 5.3 類似ユーザ情報数による性能比較

推薦タスクに使用する類似ユーザの情報数を  $k \in \{1, 2, 3, 4, 5\}$  と変化させ、その性能を比較する. Amazon Reviews の Books データセットを用い、ベクトル検索による類似ユーザ情報選択手法で比較した結果を表 3 に示す. 結果として、 $k = 3$  の場合に最も高い推薦性能が得られた. このことから、類似ユーザ情報は推薦性能向上に寄与する一方で、過度な導入は、LLM の推論における推薦対象ユーザ情報の影響を弱め、性能劣化を招くことが確認された.

## 6 おわりに

本研究では、Zero-shot 推薦において推薦対象以外のユーザ情報活用の有効性を評価した. その結果、推薦対象ユーザと嗜好が類似するユーザ情報の活用により推薦性能が向上することが示された. 一方で、類似ユーザ情報の検索方法や活用方法については、まだ改善の余地が残されている. 今後は、これらの点を改善することでさらなる性能向上を目指す.

## 参考文献

- [1] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. In **Advances in Information Retrieval: 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24–28, 2024, Proceedings, Part II**, p. 364–381, Berlin, Heidelberg, 2024. Springer-Verlag.
- [2] Lei Wang and Ee-Peng Lim. Zero-shot next-item recommendation using large pretrained language models. **CoRR**, 2023.
- [3] Seunghwan Bang and Hwanjun Song. Llm-based user profile management for recommender system. **arXiv preprint arXiv:2502.14541**, 2025.
- [4] Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. Is chatgpt a good recommender? a preliminary study. **arXiv preprint arXiv:2304.10149**, 2023.
- [5] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In **2018 IEEE International Conference on Data Mining (ICDM)**, pp. 197–206, 2018.
- [6] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In **Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM '19**, p. 1441–1450, New York, NY, USA, 2019. Association for Computing Machinery.
- [7] Jiacheng Li, Ming Wang, Jin Li, Jinmiao Fu, Xin Shen, Jingbo Shang, and Julian McAuley. Text is all you need: Learning language representations for sequential recommendation. In **Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '23**, p. 1258–1267, New York, NY, USA, 2023. Association for Computing Machinery.
- [8] Tong Zhang. Adaptrec: A self-adaptive framework for sequential recommendations with large language models. **arXiv preprint arXiv:2504.08786**, 2025.
- [9] Yu Wang, Zhiwei Liu, Jianguo Zhang, Weiran Yao, Shelby Heinecke, and Philip S Yu. Drdt: Dynamic reflection with divergent thinking for llm-based sequential recommendation. **arXiv preprint arXiv:2312.11336**, 2023.
- [10] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In **The Twelfth International Conference on Learning Representations**, 2023.
- [11] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiushi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. **arXiv preprint arXiv:2403.03952**, 2024.

**表 4** 嗜好情報および Key Feature の抽出, ユーザプロフィールの構築, ならびに推薦タスクに用いたプロンプト例.  
プロンプト

(a) 嗜好情報および Key Feature の抽出	<p>I purchased the following products and left reviews in chronological order: {Asins, product names, input reviews}</p> <p>Analyze user’s likes/dislikes/key features by referring to their reviews.</p>
(b) プロフィール構築	<p>You are given a list: {list}. Update this list by removing redundant or overlapping information. Note that crucial information should be preserved.</p>
(c) 推薦タスク	<p>{user profile}</p> <p>Reading history in chronological order: {user reading history}</p> <p>Below are the reading histories of other users with preferences similar to the target user, shown in chronological order. Each user is listed with the book they read next after the provided history.</p> <p>Other User 1: User Profile: {other user profile 1} (類似ユーザのプロフィールを使用する場合のみ) Reading History: {other user reading history 1} Next Book: {other user next book 1}</p> <p>Other User 2: User Profile: {other user profile 2} (類似ユーザのプロフィールを使用する場合のみ) Reading History: {other user reading history 2} Next Book: {other user next book 2}</p> <p>Other User 3: User Profile: {other user profile 3} (類似ユーザのプロフィールを使用する場合のみ) Reading History: {other user reading history 3} Next Book: {other user next book 3}</p> <p>Based on these inputs, rank the books in the following candidate list from 1 to 20 according to the likelihood that the target user will read them next.</p> <p>Candidate books: {candidate list}</p> <p>When ranking the candidate books, prioritize the target user’s own preferences and recent reading history. Treat the next book from each other user as a signal of potential future interest. Focus on personal preference alignment rather than general popularity.</p>

## A プロンプト例

レビューからの嗜好情報および Key Feature の抽出, ユーザプロフィール構築, ならびに推薦タスクに用いた各プロンプトを, それぞれ表 4 (a), (b), (c) に示す. なお, 推薦タスクのプロンプトには, Amazon Reviews の Books データセットを対象とした例を用いる.