

Sequential Recommendation におけるテキスト情報を活用した未知アイテムへの対処法に関する分析

深澤祐 山口泰弘
クックパッド株式会社

{yusuke-fukasawa,yasuhiro-yamaguchi}@cookpad.com

概要

Sequential Recommendation Model は履歴中に出現するアイテムの順序情報を元にして次にアクションし得るアイテムを推薦する実世界の応用に適したモデルである。このモデルは未知ユーザであっても既知アイテムで構成された履歴を持つならば推薦が可能である一方で、未知のアイテムが入力として与えられた場合そのアイテムを用いた推薦は多くの場合で困難である。本研究では、モデルの種類に依存しない手法としてテキスト情報を活用して未知アイテムの ID 埋め込みを推測する方法を検討した。実験結果からは、テキスト情報に基づく推測手法は限定的ながらも一定の効果を示した。また、類似度が高いアイテムによる置換や未知アイテムを入力履歴から除外することで推薦精度を保つことが確認された。

1 はじめに

情報技術の発展に伴いユーザーの興味やニーズに合わせた情報を効果的に推薦するシステムの必要性が増している。その中でも Sequential Recommendation Model は、ユーザの行動履歴におけるアイテムの並びから次のアイテムを予測するというタスクを解いており、実世界の動的なユーザ行動に基づいた推薦に適したモデルとして知られている。Sequential Recommendation Model には様々なモデルが提案されており、BERT [1] を用いてアイテムを埋め込む手法として BERT4Rec [2]、ユーザの情報を組み込んだ階層的なモデルである SHAN [3] などが提案されている。これらのモデルは学習時に存在していないユーザであっても、既知アイテムで構成された履歴さえあれば推薦が可能であるという点が特徴的である。一方でアイテムの埋め込み表現 (ID Embedding) を持たない場合そのアイテムに対する値を算出することができないため、推薦を行うことはできない。つまり

システムに新しく追加された未知アイテムを用いた推薦は困難である。現実のアプリケーションでは新しいアイテムが頻繁に登録され行動履歴中に出現することが多いため、未知アイテムを適切に取り扱うことは Sequential Recommendation Model における重要なトピックである。

2 関連研究

このような未知アイテムに着目した研究として、メタ学習を用いたアプローチが提案されている。Zheng ら [4] は Meta-learning-based Cold-Start Sequential Recommendation Framework: Mecos を提案している。Mecos では限られたインタラクションデータからユーザの嗜好を抽出し、インタラクションの少ない Cold-Start アイテムとユーザとのマッチングを学習することで、Cold-Start アイテムを使った推薦の精度を向上させる手法を提案している。Pan ら [5] はテキスト及び画像の特徴量を合わせた埋め込み表現を用いて学習するマルチモーダルメタ学習を提案した。Dong ら [6] はタスク固有の重みを学習するメモリと特徴についての重みを学習するメモリの2つを格納できるメモリ行列を提案し、大域的な特徴と局所的な特徴の両方を捉えられるようにすることで Cold-Start な状況でも機能する推薦モデルを構築した。

また自然言語処理の観点で見れば、未知アイテムは OOV (Out of Vocabulary) に対処する問題として捉えられる。Chen ら [7] は BERT などの事前学習モデルの埋め込み表現を元にしてサブワード的なアプローチにより未知語の埋め込み表現を推測する手法を提案している。単語の表層情報を使わずに文脈情報を用いて未知語の埋め込み表現を推定する方法として深層学習を用いた手法も提案されている [8]。

本論文では Sequential Recommendation Model において学習データに含まれない未知アイテムをどのよ

うに取り扱うべきかについての分析を行う。ID の他にテキストデータを情報として持つデータを対象として推薦モデルを学習した上で、その推薦モデル内の ID Embedding をテキストデータから推測する手法を提案し、この手法が他の未知アイテムを取り扱う手法に対してどの程度優位性を備えているかを検証する。

3 問題設定

本論文ではクックパッド株式会社が提供しているレシピアプリ「クックパッド」のレシピデータを用いて実験を行う。レシピアプリ「クックパッド」において、ユーザは検索行動などを経てレシピを閲覧し、実際にレシピを作った際にはそれを伝えるメッセージ(つくれば)を送ることができる。本論文ではこの閲覧履歴とつくればの投稿履歴を用いて推薦を行う状況を想定する。異なる性質のデータを対象とした推薦を行うシチュエーションにおける未知アイテムの取り扱い方法について比較的分析を行う。

実験に利用する Sequential Recommendation Model として、今回は CORE [9] を採用した。CORE はアイテム ID が連なった配列をセッションとしたとき、そのセッションの埋め込み表現を ID Embedding から獲得した後、セッションの埋め込み表現とその次にアクションしたアイテムの埋め込み表現を近づけるように学習する一方で、アクションしていないアイテムの表現とを離すように学習し精緻なセッションの埋め込み表現を構築する。この埋め込み表現を用いて推薦が行われる。[9] では GRU4Rec [10] や SASRec [11] といった主要なモデルとの比較が行われた上でそれらよりも多くのデータセットにおいて精度が高いことが示されている。

本論文ではメタ学習的な手法で新たに別の情報を組み込んだモデルを提案するのではなく、こうした既存のモデルにおいて未知アイテムを適切に取り扱う手法を提案する。具体的には学習が完了したある推薦モデルについて、そのモデルが獲得したアイテムの ID Embedding をテキストデータから推測する。このような推論を行うモデルを学習させることで、未知アイテムに対してはそれに紐づくテキストデータを入力として推測した ID Embedding を用いて推薦を行うことが可能となる。この手法を Embedding Mapping と呼称する。この手法に対するより簡便な比較手法として Ignore 及び Replace by Similarity を提案する。Ignore は未知アイテムを無視する手法であ

	Recipe Tsukurepo	Recipe Detail View
ユーザ数	125,946	374,091
アイテム数	206,186	387,227
レコード数	1,000,000	3,590,206

表 1 実験で用いたデータセット

データセット	NDCG@50
Recipe Tsukurepo	0.029
Recipe Detail View	0.3628

表 2 各データセットで CORE を学習した後テストデータで評価した際の nDCG@50

り、Replace by Similarity は未知アイテムをテキストの類似度が最も高い別の既知アイテムに置き換える。これらの手法を用いて推薦を行った際の精度を比較することで、Embedding Mapping の有効性を検証する。

4 実験

実験で用いたデータセットの統計を表 1 に示す。表内の Recipe Tsukurepo がつくれば投稿履歴のデータ、Recipe Detail View が閲覧履歴のデータである。なお、クックパッド株式会社が研究用途として国立情報学研究所を經由して公開しているレシピデータセット [12] では、“つくれば” データは格納されているが閲覧データは存在しない。そのため本論文で用いる閲覧データは公開されているレシピデータセットではなく、「クックパッド」アプリの運用から得られたデータを用いる。

このデータについて、各ユーザの履歴を時系列順に並べた際の先頭から 80% を学習データ、次の 10% を評価データ、残りの 10% をテストデータとした。テストデータについては履歴の長さが 10 以上のものを対象とした。また評価指標として、本論文では推薦で得られるランキングに対する評価指標として nDCG [13] を用いる。CORE の ID Embedding のサイズを 64 として学習を行い、テストデータに対して評価した際の nDCG@50 を表 2 に示す。なおモデルの学習には RecBole を利用した [14] [15]。

得られた推薦モデルから ID Embedding を取り出し、テキストデータは事前に学習した fastText および BERT を用いてベクトル表現に変換する。予め各セッションごとのレシピのタイトル列を元に学習した fastText [16] 及び BERT を用意し、fastText を用いるものはその出力を LSTM [17] に接続してアイテムのテキストデータに対応する ID Embedding に変換す

未知アイテム数	履歴先頭から		履歴末尾から	
	LSTM	BERT	LSTM	BERT
1	0.3637	0.3610	0.3275	0.3221
2	0.3600	0.3615	0.2888	0.2616
3	0.3608	0.3613	0.2223	0.1545
4	0.3574	0.3616	0.1406	0.0689
5	0.3565	0.3573	0.0841	0.0331

表 3 Recipe Detail View に対する推薦モデル: 履歴先頭と末尾からの Embedding 置き換え時の nDCG@50 の比較

未知アイテム数	履歴先頭から		履歴末尾から	
	LSTM	BERT	LSTM	BERT
1	0.0294	0.0290	0.0282	0.0279
2	0.0289	0.0291	0.0271	0.0274
3	0.0291	0.0286	0.0259	0.0259
4	0.0287	0.0289	0.0251	0.0257
5	0.0284	0.0284	0.0247	0.0239

表 4 Recipe Tsukurepo に対する推薦モデル: 履歴先頭と末尾からの Embedding 置き換え時の nDCG@50 の比較

るようモデルを学習する。これを LSTM Embedding Mapping と呼称する。BERT については分かち書きを行った後のトークンについての出力を平均したものをレシピに関する表現とし、その後に全結合層を通して Embedding を予測するように学習を行った。これを BERT Embedding Mapping と呼称する。

未知アイテムに関する設定について、推薦モデルに入力するアイテムの履歴で先頭または末尾から順に最大 5 つが未知のアイテムであり、ID Embedding が取得できずモデルに入力できない状態だと仮定する。このとき、履歴の長さが 10 以上のものを対象とした。Embedding Mapping の場合はこれらの Embedding をテキストデータから予測し、これを CORE の Embedding と置き換えた上でその後の処理を行い推薦を行う。Ignore の場合はこれらの未知アイテムを履歴から除外したものを新しい履歴として推薦モデルに入力する。Replace by Similarity は予めクックパッドのレシピタイトルで学習した fastText を用いて、未知アイテムのタイトルと最も cos 類似度が高い既知のアイテムを選択し、未知アイテムとなっている部分をそのアイテムで置き換える処理を行う。得られた新しい履歴を推薦モデルに入力する。

まず LSTM Embedding Mapping と BERT Embedding Mapping のそれぞれを用いて推薦を行った際の nDCG@50 を表 3, 4 に示す。Recipe Detail View において履歴先頭から未知アイテムを仮定した場合は

BERT Embedding Mapping の方が概ね良好な結果を示したのに対して、末尾から未知アイテムを仮定した場合の結果では LSTM Embedding Mapping が全体的に優位な結果を示した。Recipe Tsukurepo においてはどちらの場合でもはっきりとした優劣がつけられる結果が得られなかった。なお未知アイテムが存在した際に精度へ与える影響を 2 と比較しつつ見ていくと、先頭に未知アイテムが存在する場合はそもそも影響が小さいこともあり nDCG@50 はあまり減少していない。これに対して履歴末尾が未知アイテムであると仮定した場合は nDCG@50 が大きく減少している。これより CORE がより直近の履歴が予測結果に強い影響を与えていることが推察される。

次に Embedding Mapping と Ignore, Replace by Similarity についての比較を行う。ここでは LSTM Embedding Mapping を比較対象として用いた。結果を表 5, 6 に示す。Recipe Detail View において、履歴先頭から未知アイテムを仮定した場合は Mapping よりも Ignore, Replace の精度が高く、履歴末尾から未知アイテムを仮定した場合は Replace by Similarity を行うことで nDCG@50 の減少を抑えられた。Recipe Tsukurepo において、どちらの方向から未知アイテムを仮定する場合でも概ね Ignore が良好な結果を示した。

5 結果に対する分析

まず本論文で提案した Embedding Mapping が他手法を下回る結果を出した点について考察する。LSTM Embedding Mapping を学習した際の RMSE 損失は Recipe Detail View を対象とした場合で 0.0112、Recipe Tsukurepo を対象とした場合で 0.0057 であった。それぞれの場合で得られるベクトルの性質が CORE 内の ID Embedding の性質と一致するかを確認するために、fastText 埋め込み表現上で cos 類似度が 0.8 以上のタイトルを持つレシピの組を 100 組抽出し、それらについて LSTM Embedding Mapping が出力するベクトル同士での cos 類似度平均と、CORE 内の ID Embedding 同士での cos 類似度平均を比較した。得られた結果を表 7 に示す。この結果から今回学習した LSTM Embedding Mapping は CORE 内の ID Embedding とは異なり、fastText の埋め込み表現の傾向と似た傾向を示している。この結果は BERT を用いたモデルの場合でも概ね同様の結果を示した。このことから Embedding Mapping のモデルについては CORE 内の ID Embedding との乖離が大きく改善す

未知アイテム数	履歴先頭から			履歴末尾から		
	Ignore	Replace	Mapping	Ignore	Replace	Mapping
1	0.3606	0.3631	0.3637	0.3310	0.3615	0.3275
2	0.3606	0.3590	0.3600	0.3041	0.3565	0.2888
3	0.3674	0.3639	0.3608	0.2960	0.3597	0.2223
4	0.3600	0.3619	0.3574	0.2798	0.3603	0.1406
5	0.3600	0.3625	0.3565	0.2733	0.3577	0.0841

表 5 Recipe Detail View に対する推薦モデル: 履歴先頭と末尾における未知アイテムに対する処理方法による nDCG@50 の比較

未知アイテム数	履歴先頭から			履歴末尾から		
	Ignore	Replace	Mapping	Ignore	Replace	Mapping
1	0.0294	0.0291	0.0294	0.0286	0.0271	0.0282
2	0.0293	0.0285	0.0289	0.0273	0.0262	0.0271
3	0.0284	0.0279	0.0291	0.0265	0.0243	0.0259
4	0.029	0.0268	0.0287	0.0262	0.0230	0.0251
5	0.0289	0.0261	0.0284	0.0255	0.0207	0.0247

表 6 Recipe Tsukurepo に対する推薦モデル: 履歴先頭と末尾における未知アイテムに対する処理方法による nDCG@50 の比較

データセット	fastText	CORE	Mapping
Recipe Detail View	0.8919	0.2838	0.9555
Recipe Tsukurepo	0.8752	0.0668	0.9899

表 7 タイトル類似ペアを対象とした各モデルの埋め込みベクトルに関する平均 cos 類似度の比較

る余地があると考えられる。

次に Replace by Similarity について、この手法は Recipe Detail View の方がより効果を発揮した。これについてはデータの性質が大きく関係していると思われる。Recipe Detail View の履歴におけるレシピのタイトル列について時間的に隣接する 2 つのレシピの cos 類似度の平均を計算したところ、その値が 0.6867 だったのに対して、Recipe Tsukurepo で同様に計算した cos 類似度平均値は 0.4817 であった。Recipe Detail View は検索行動を通じて得られた履歴であるため短期的な興味を反映して似ているレシピを閲覧する傾向が強い。これに対し Recipe Tsukurepo の場合はより長期的な興味の推移が現れる履歴になっていると言える。類似しているアイテムが連続しやすいデータにおいては未知アイテムを別の似ているアイテムで置き換えることで履歴の性質を大きく変えることなく補完できたのだと考えられる。

最後に Ignore が多くの場合で良好な結果を示した点について、今回対象としたテストデータが履歴の長さが 10 以上のものを対象としたことが影響していると考えられる。特に履歴先頭の方で未知アイテム

が存在している場合、CORE が注目しているのは履歴末尾の直近接触したアイテムであると考えられるため影響が軽微である。そのため、他手法のように別のアイテムや Embedding に置き換えるよりも未知アイテムを除外してしまう方が一番影響を抑えられたのだと考えられる。

Sequential Recommendation Model を運用する上で未知アイテムが履歴中に出現し得る場合、その出現位置やデータセットの性質に応じた手法を用いることが重要であることが今回の結果より示された。

6 おわりに

本論文では Sequential Recommendation Model を運用する際注意すべき未知アイテムの取り扱い方法について、テキスト特徴量からその ID Embedding を推測する手法 Embedding Mapping を提案した。Embedding Mapping は優れた手法とは言えない結果ではあったが、Ignore や Replace by Similarity といったシンプルな手法でモデルの種類にかかわらず未知アイテムがあったとしても推薦精度をそれほど落とさないことが限定的ながらも示された。今後 Embedding Mapping 手法の改善に努めると共に、履歴中に未知アイテムが存在するパターンについてランダムな位置に未知アイテムが出現する場合といった実際の状況に即した条件での影響を調べるなどを通じてより精緻な分析を行っていく。

参考文献

- [1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In North American Chapter of the Association for Computational Linguistics, 2019.
- [2] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. Proceedings of the 28th ACM International Conference on Information and Knowledge Management, 2019.
- [3] Haochao Ying, Fuzhen Zhuang, Fuzheng Zhang, Yanchi Liu, Guandong Xu, Xing Xie, Hui Xiong, and Jian Wu. Sequential recommender system based on hierarchical attention networks. In International Joint Conference on Artificial Intelligence, 2018.
- [4] Yujia Zheng, Siyi Liu, Zekun Li, and Shu Wu. Cold-start sequential recommendation via meta learner. In AAAI Conference on Artificial Intelligence, 2020.
- [5] Xingyu Pan, Yushuo Chen, Changxin Tian, Zihan Lin, Jinpeng Wang, He Hu, and Wayne Xin Zhao. Multimodal meta-learning for cold-start sequential recommendation. Proceedings of the 31st ACM International Conference on Information & Knowledge Management, 2022.
- [6] Manqing Dong, Feng Yuan, Lina Yao, Xiwei Xu, and Liming Zhu. Mamo: Memory-augmented meta-optimization for cold-start recommendation. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020.
- [7] Lihu Chen, Gaël Varoquaux, and Fabian M. Suchanek. Imputing out-of-vocabulary embeddings with love makes languagemodels robust with little cost. ArXiv, Vol. abs/2203.07860, , 2022.
- [8] Johannes V. Lochter, Renato Moraes Silva, and Tiago A. Almeida. Deep learning models for representing out-of-vocabulary words. In Brazilian Conference on Intelligent Systems, 2020.
- [9] Yupeng Hou, Binbin Hu, Zhiqiang Zhang, and Wayne Xin Zhao. Core: Simple and effective session-based recommendation within consistent representation space. Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2022.
- [10] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. CoRR, Vol. abs/1511.06939, , 2015.
- [11] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. 2018 IEEE International Conference on Data Mining (ICDM), pp. 197–206, 2018.
- [12] クックパッド株式会社. クックパッドデータ, feb 2015.
- [13] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. ACM Trans. Inf. Syst., Vol. 20, pp. 422–446, 2002.
- [14] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. In CIKM, pp. 4653–4664. ACM, 2021.
- [15] Lanling Xu, Zhen Tian, Gaowei Zhang, Junjie Zhang, Lei Wang, Bowen Zheng, Yifan Li, Jiakai Tang, Zeyu Zhang, Yupeng Hou, Xingyu Pan, Wayne Xin Zhao, Xu Chen, and Ji-Rong Wen. Towards a more user-friendly and easy-to-use benchmark library for recommender systems. p. 2837–2847, 2023.
- [16] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606, 2016.
- [17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural Computation, Vol. 9, No. 8, pp. 1735–1780, 1997.