M2SR: Multi-Modal Multi-Fusion Architecture for Sequential Recommendation

Jiawei Huang 30920231154349[1], Chaoran Wang 30920231154361[2], Qiyue Wu 36920231153242[2], Yibo Xie 36920231153201[2]

[1]School of Informatics Xiamen University
[2]Xiamen University Artificial Intelligence Research Institute
wangduphunsukh@stu.xmu.edu.cn, 22920192204281@stu.xmu.edu.cn,
huangjiawei@stu.xmu.edu.cn, wuqiyue@stu.xmu.edu.cn

Abstract

Sequential recommendation aims to offer potentially interesting products to users by capturing their historical sequence of interacted items. With the emergence of multimedia services, such as short video, news and etc., understanding these contents while recommending becomes critical. Multi-modal data that depicts a user's historical interactions exists ubiquitously, such as product pictures, textual descriptions, and interacted item sequences, providing semantic information from multiple perspectives that comprehensively describe a user's preferences. However, existing sequential recommendation methods either fail to directly handle multi-modality or suffer from high computational complexity. To address this, we propose a novel Multi-Modal Multi-Fusion Architecture for Sequential Recommendation(M2SR). It is a MLP-based architecture that consists of two modules - the Feature Fusion Layer and Prediction Layer - and has an edge on both efficacy and efficiency. Extensive experiments show that the multi-modal representation learned by our proposed model generally benefits other recommendation models. Thus, the Feature Fusion Layer proposed in our scheme can be applied to enhance other recommendation models, thereby contributing to their significant improvement.

Introduction

With the rapid growth of e-commerce, users often find themselves overwhelmed by a multitude of trendy content, and over time, their preferences undergo dynamic shifts. Capturing this evolving preference has become a paramount task for content providers.(Liu et al. 2016) Sequential Recommendation Systems (SRS) gain a significant edge in depicting how user behavior changes over time by modeling users' historical interaction records and recommending items they may interact with in the future. SRS plays a pivotal role in contemporary life, spanning applications in search engines, advertising systems, e-commerce platforms, video and music streaming services, as well as various other online platforms.

In recent years, the swift progress in deep learning has given rise to a variety of deep learning-based sequential recommendation models.(Hidasi et al. 2015) Notably, two predominant approaches have emerged: those based on Recurrent Neural Networks (RNNs)(Hidasi et al. 2015) and those employing self-attention mechanisms(Kang and McAuley 2018; Zhang et al. 2019). RNNs are traditionally perceived as highly effective in handling sequentially related data. Nevertheless, even though they have achieved advanced performance(Kang and McAuley 2018; Zhang et al. 2019), be it through Long Short-Term Memory (LSTM)(Hochreiter and Schmidhuber 1997) or Gated Recurrent Units (GRU)(Cho et al. 2014), they still grapple with challenges related to sustaining long-term dependencies and parallel processing. Self-attention(Vaswani et al. 2017), a burgeoning concept, operates without these constraints, enabling the capture of long-term relationships between items without depending on their relative positions. Self-attention has achieved state-of-the-art levels(Kang and McAuley 2018; Zhang et al. 2019).

While existing research(Li et al. 2022; Zhang et al. 2019) has underscored the use of side information to accurately model users' sequential behavior, there has been limited exploration of multimodal sequential recommendations. User sequential behavior is seldom considered as multimodal. However, in the realm of recommendation systems, multimodal data is gaining increasing attention as it provides semantic information from various perspectives of user interactions. For instance, traditional sequential recommendation systems may struggle to capture semantic information from item images or textual descriptions, which can be crucial for users interested in specific colors or types of vehicles. To address this challenge, it is imperative to derive latent embeddings from different representations of items.

Related Work

ID-based recommender systems (IDRec)(Koren, Bell, and Volinsky 2009) have received much attention in the recommendation literature. They can be roughly divided into two categories: non-sequential models (NSM) and sequential neural models (SRM). NSM includes various recall models (e.g. DSSM, YouTube DNN(Covington, Adams, and Sargin 2016)) and CTR models (e.g. DeepFM(Guo et al. 2017), Wide &Deep(Cheng et al. 2016), Deep Crossing(Shan et al. 2016)). These models take user-item pairs as input along with features to predict matching scores. In contrast, SRM takes sequences of user-item interactions as input to generate next interaction probabilities. Representative SRM includes GRU4Rec(Hidasi et al. 2015), NextItNet(Yuan et al. 2021), SR-GNN(Wu et al. 2019), SAS-

Rec(Kang and McAuley 2018) and BERT4Rec(Sun et al. 2019) with RNN, CNN, GNN, Transformer and BERT as the backbones,respectively, among which SASRec often performs the best(Yuan et al. 2022).

Modality-based recommender systems (MoRec) mainly focus on modeling the content features of different modalities such as text(Wu et al. 2020), images(McAuley et al. 2015), videos(Deldjoo et al. 2016), audio(Van den Oord, Dieleman, and Schrauwen 2013) and multimodal text-image pairs(Wu et al. 2021). Previous work tended to adopt a two-stage (TS) mechanism by first pre-extracting fixed item modality features and then incorporating them into the recommendation model. Most of these works use modality as side features and IDs as main features. End-to-end (E2E) MoRec has gained popularity recently due to: (1) availability of high-quality public datasets with original item modalities; (2) advances in modality encoders (ME) like word embeddings. However, most existing E2E MoRec focus on text recommendation(Hou et al. 2022).

Recent studies have shown MLP-based architectures demonstrate superior performance in computer vision (CV) and natural language processing (NLP), comparable to mainstream Transformers. In CV, representatives include MLP-Mixer(Tolstikhin et al. 2021), resMLP(Touvron et al. 2022), etc. In NLP, pNLP-Mixer(Fusco, Pascual, and Staar 2022)achieve similar functionality to self-attention using token mixing and input weighted summation. For sequential recommendations, FMLP-REC(Zhou et al. 2022) and MLP4Rec(Li et al. 2022) pioneered MLP-based models, though they are not yet widely used for multimodal sequence recommendation. Our proposed model provides an effective MLP-based solution for this task.

Solution

A typical multi-modal sequential recommendation system, as shown in Figure 1, incorporates both the user's short-term and long-term preferences through the display of interaction history and sequence information. By leveraging these details, the multi-modal sequential recommendation system analyzes user preferences to provide recommendations for relevant items. Unlike item IDs that only reveal partial sequential patterns, the multi-modal feature sequence offers a more comprehensive view of underlying patterns. Consequently, it has become increasingly common to integrate item features using models based on recurrent neural networks (RNNs) and self-attention mechanisms in order to incorporate multi-modal features into sequential recommendations (Zhang et al. 2019). However, RNNs fall short in capturing long-term dependencies, and attention mechanisms are computationally expensive.

To address the aforementioned issues, we propose a novel method for sequential recommendation that effectively captures and integrates multi-modal information to generate informative predictions for the next item. Our approach consists of two key components: the Feature Fusion Layer and the Prediction Layer. The Feature Fusion Layer includes three fusion modules that intricately capture and blend the multi-modal representations of multiple items, leveraging attention mechanisms and graph neural networks to capture dependencies and relationships between different modalities. This fusion process results in a comprehensive representation that is then passed to the Prediction Layer, which generates the next item recommendation.

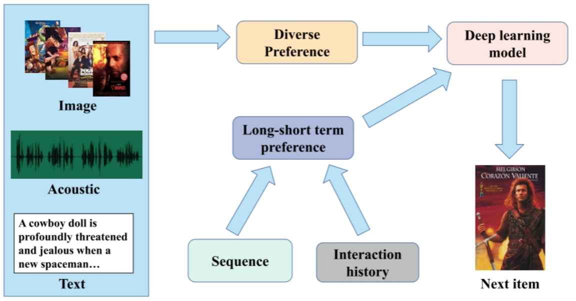


Figure 1: The general paradigm of multi-modal sequential recommender system

To validate the effectiveness of our proposed method, we plan to evaluate it on benchmark datasets, namely Yelp and Movielens 1M. Through extensive experiments and comprehensive analyses, we aim to demonstrate the superiority of our approach over existing baseline sequential recommendation methods and competitive auxiliary information integration methods on these datasets. Additionally, we envision that the Feature Fusion Layer proposed in our scheme can be applied to enhance other recommendation models, thereby contributing to their significant improvement.

In summary, our proposed solution entails the following aspects:

- We propose a novel multi-modal sequential recommendation approach that integrates and aligns multi-modal information in sequential recommendations, effectively capturing users' fine-grained preferences.
- We conduct extensive experiments to validate the effectiveness of our proposed method and perform comprehensive analyses to verify the efficacy of each component.
- We aim to enhance the compatibility of our solution and explore the application of the proposed Feature Fusion Layer to enhance other recommendation models, while conducting relevant validations.

By addressing the limitations of existing sequential recommendation methods, our research contributes to the advancement of multi-modal sequential recommendation systems, leading to more accurate and personalized recommendations across various domains. The proposed approach and its empirical evaluations provide valuable insights for researchers and practitioners in the field of recommendation systems.

Framework

In this paper, we propose a multi-modal multi-fusion Architecture for sequential recommendations(M2SR) that can explicitly learn information from various modalities. The framework consists of three layers: the Feature Mixer Layer, Fusion Mixer Layer, and Prediction Layer.

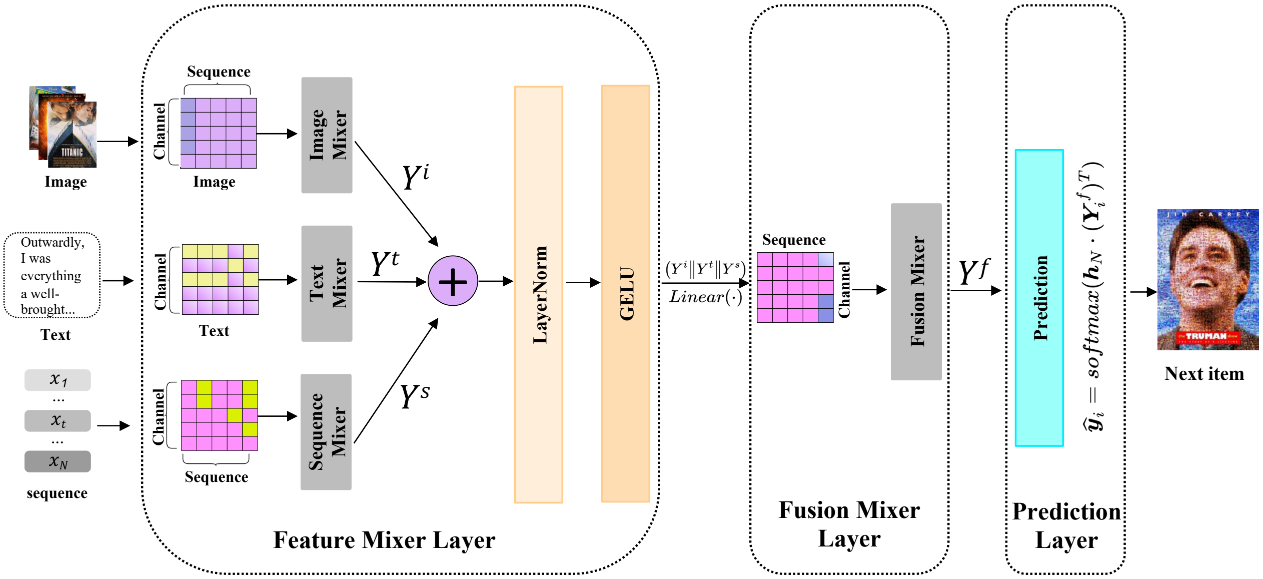


Figure 2: The framework overview of the proposed M2SR.

Our framework is flexible and can incorporate data in diverse modalities. We focus on images and texts in this paper as they are the most commonly used modalities in addition to item sequences. As shown in Figure 2, image, text and item sequences from the user-item interaction history are used as input. We design three Mixer Modules in the Feature Mixer Layer to extract and process image, text, and item sequence information respectively. The Feature Mixer Layer also includes layer normalization and residual connections to enhance training stability. Next, we adopt a post-fusion approach in the Fusion Mixer Layer by concatenating the outputs $\hat{Y^i}$, Y^t,and Y^s from the three Mixer Modules to fuse the representations from multiple modalities. Finally, we make next item recommendations in the Prediction Layer based on the fused representation.

The Feature Mixer Layer contains three Mixer Modules to extract image, text, and item sequence information respectively. We first encode the multi-modality raw data into embedding feature matrices. Specifically, we load images as a feature matrix, utilize a pretrained model to encode text, and set trainable embeddings for the item sequence. Next, the three embedded inputs I, T, and S from images, text, and item sequences are fed into the Mixer Modules for processing. As depicted in Figure 3, each Mixer Module comprises identical blocks, and each block performs two mixing operations. We take image modality feature matrix I as an example, while operations on text T and item sequence S are identical. The first mixing is token mixing with a token size of D_I. The token mixer, denoted as TM, acts on the columns of I to capture token-level interactions within each channel. The output is then passed to a channel mixer CM, which operates on the rows of I to capture channel-level interactions across tokens. Standard components including residual connections and layer normalization are utilized to stabilize training. For simplicity, the Mixer Module operations on the image feature matrix I can be denoted as:

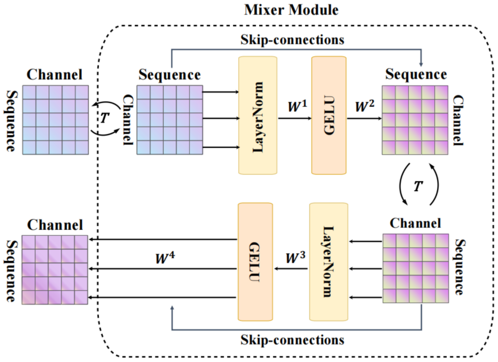


Figure 3: The detailed architecture of Mixer Module.

$$\widehat{I}_{*,i} = I_{*,i} + TM\left(LayerNorm(I)_{*,i}\right), \quad \text{for } i = 1 \dots D_I,$$

$$\widehat{I}_{j,*} = \widehat{I}_{j,*} + CM\left(LayerNorm(\widehat{I})_{j,*}\right), \quad \text{for } j = 1 \dots N, \tag{1}$$

where $I_{*,i}$ represents the operations on the column dimension, i.e. cross-token processing, on the image matrix, and

$I_{j,*}$ is the operations on the row dimension, i.e. cross-channel processing. $\widehat{I}$ denotes the intermediate representation for the image modality. Through the same process on T and S , we can obtain the intermediate text representation $\widehat{T}$ and sequence representation $\widehat{S}$.

We propose the Fusion Mixer Layer to fuse the representations from multiple modalities. A post-fusion approach is adopted by concatenating the outputs Y^i, Y^t, and Y^s from all the Mixer Modules and feeding them into the Fusion Mixer Layer, which contains another mixer module. By fusing multi-modal representations through the Fusion Mixer Layer, we can obtain a comprehensive representation of the user's historical item interaction sequence. The output of the Fusion Mixer Layer is formulated as:

$$\widehat{Y}_{*,i} = \widehat{Y}_{*,i} + W^{14}\sigma\left(W^{13} LayerNorm(\widehat{Y})_{*,i}\right), \text{for } i = 1 \ldots D, \tag{2}$$

where $\widehat{Y}$ = Linear$(Y^i||Y^t||Y^s)$ and is the concatenation operation, so $D = D_I + D_T + D_S$. Y^f is the output of the block, which is the comprehensive representation considering multiple modalities $W^{13} \in R^{r_N \times N}$ and $W^{l4} \in R^{N \times r_N}$ denote the learnable weights of the first layer in the mixer. $W^{15} \in R^{r_D \times D}$ and $W^{16} \in R^{D \times r_D}$ are the learnable weights of the second layer in the mixer.

We present the optimization algorithm for our proposed model in Algorithm 1. We first randomly initialize the model parameters (line 1). In each epoch, the training data is split into batches (line 3). The feature matrices of the three modalities X, T and S are then fed into the token mixers TM and channel mixers CM to obtain the intermediate representations x, t and s respectively (line 4). Based on the image mixer (line 5), text mixer (line 6) and sequence mixer (line 7), we can generate the representations YI, YT and YS corresponding to the three modalities. We fuse the multi-modal features to get the comprehensive representation Yf based on the Fusion Mixer Layer (line 8). The loss is calculated and model parameters are updated via gradient descent until convergence (line 9). Notably, the image mixer, text mixer and sequence mixer only involve simple matrix multiplications, thus preserving the linear time complexity.

Experiment

We evaluate our model on two widely used benchmark datasets - MovieLens 100K and MovieLens 1M[2]. The number of interactions and average sequence length are 99,287 (105) and 999,611 (165), respectively. We filter out items and users with less than 5 interactions. The maximum sequence length is set as 50 for both datasets, with zero padding for shorter sequences. For data splitting, the last item in each interaction sequence is used as the test set, the second last item as the validation set, and the remaining items as the training set.

We evaluate the efficacy based on next-item prediction. Two commonly used evaluation metrics for recommender systems are adopted: mean reciprocal rank (MRR) and normalized discounted cumulative gain (NDCG). MRR considers the rank of the ground-truth item in the top-K recommendation list, then averages it over all test instances. We report the average MRR over 3 random seeds. NDCG measures the ranking quality of the top-K recommendations generated by the model. It assigns higher scores to ground-truth items appearing earlier in the ranked list.

Algorithm 1: Optimization pipeline of M2SR

Input: Historical interacted item feature matrix of image modality: I, text modality: T, item sequence: S
Output: Well-train model f_θ

```
 1: Randomly initialize parameters of model $f_\theta$.
 2: for Epoch in 1,...,max epochs do
 3:     for Batch in 1,...,batch number do
 4:         Generate the intermediate representation of three
            modalities $\widehat{I}, \widehat{T}, \widehat{S}$ by Eq. (1).
 5:         Learn image representation $Y^i$ by Eq. (2).
 6:         Learn text representation $Y^t$ by Eq. (1).
 7:         Learn sequence representation $Y^s$ by Eq. (1).
 8:         Fuse the multi-modal features and get the compre-
            hensive representation $Y^f$ by Eq. (1).
 9:         Calculate the CrossEntropy loss and update $\theta$.
10:     end for
11: end for
12: return Well-trained model $f_\theta$.
```

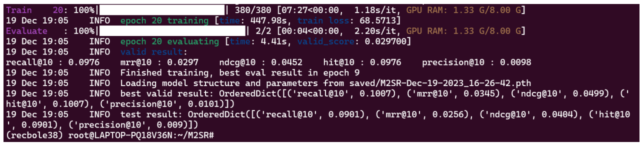
Baselines. We compare the performance of our proposedMMMLP model against several widely used baselines in the field of recommender systems. These baselines include:FPMC, BPR, GRU4Rec, SASRec and MLPMixer.

- **FPMC**: FPMC combines Markov Chains and Matrix Fac-torization method to learn the sequential dependencies in userinteraction history as well as users'general preferences.
- **BPR**: BPR builds matrix factorization model from pairwiseloss function to learn from implicit feedback, and it is a classicalgeneral recommender system.
- **GRU4Rec**: GRU4Rec uses gated recurrent unit to improvethe performance of vanilla RNN, allowing it to mitigate the van-ishing gradient problem to some extent.
- **SASRec**: A sequential recommendation model based onattention that uses a self-attention network for the generation ofsequential recommendations.
- **MLPMixer**: MLPMixer is our improved version of MLP-Mixer to make it adapt to sequential recommendation tasksbased on item embeddings.

Our M2SR implementation and all baseline models are built on the open-source recommendation library RecBole. This provides a fair and reproducible environment for comparing different methods. Hyperparameters are set according to the values reported in the original papers. The Adam optimizer and early stopping are used for training. When detailed hyperparameters are unavailable in the papers, we tune them via cross-validation.

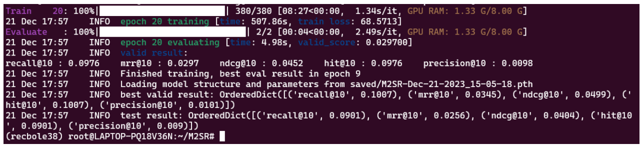
We set the learning rate to 1e-4, and fix the batch size as 256.Moreover, to handle different item sequence lengths, we use paddingto fill users whose interaction numbers are

Dataset	Metric	FPMC	BPR	GRU4Rec	SASRec	MLPMixer
ML-100K	MRR@10 NDGG@10	0.1314 0.1932	0.1513 0.2132	0.1829 0.2521	0.1909 0.2704	0.2144 0.2636
ML-1M	MRR@10 NDGG@10	0.2419 0.3040	0.2959 0.3535	0.3383 0.4062	0.4043 0.4430	0.4129 0.4695



```
Train    20: 100%|██████████████████████| 380/380 [07:27<00:00,  1.18s/it, GPU RAM: 1.33 G/8.00 G]
19 Dec 19:05    INFO  epoch 20 training [time: 447.98s, train loss: 68.5713]
Evaluate   : 100%|████████████████████████████| 2/2 [00:04<00:00,  2.20s/it, GPU RAM: 1.33 G/8.00 G]
19 Dec 19:05    INFO  epoch 20 evaluating [time: 4.41s, valid_score: 0.029700]
19 Dec 19:05    INFO  valid result:
recall@10 : 0.0976    mrr@10 : 0.0297    ndcg@10 : 0.0452    hit@10 : 0.0976    precision@10 : 0.0098
19 Dec 19:05    INFO  Finished training, best eval result in epoch 9
19 Dec 19:05    INFO  Loading model structure and parameters from saved/M2SR-Dec-19-2023_16-26-42.pth
19 Dec 19:05    INFO  best valid result: OrderedDict([('recall@10', 0.1007), ('mrr@10', 0.0345), ('ndcg@10', 0.0499), ('
hit@10', 0.1007), ('precision@10', 0.0101)])
19 Dec 19:05    INFO  test result: OrderedDict([('recall@10', 0.0901), ('mrr@10', 0.0256), ('ndcg@10', 0.0404), ('hit@10
', 0.0901), ('precision@10', 0.009)])
(recbole38) root@LAPTOP-PQ18V36N:~/M2SR#
```

Figure 4: Test result in ML-100K



```
Train    20: 100%|██████████████████████| 380/380 [08:27<00:00,  1.34s/it, GPU RAM: 1.33 G/8.00 G]
21 Dec 17:57    INFO  epoch 20 training [time: 507.86s, train loss: 68.5713]
Evaluate   : 100%|████████████████████████████| 2/2 [00:04<00:00,  2.49s/it, GPU RAM: 1.33 G/8.00 G]
21 Dec 17:57    INFO  epoch 20 evaluating [time: 4.98s, valid_score: 0.029700]
21 Dec 17:57    INFO  valid result:
recall@10 : 0.0976    mrr@10 : 0.0297    ndcg@10 : 0.0452    hit@10 : 0.0976    precision@10 : 0.0098
21 Dec 17:57    INFO  Finished training, best eval result in epoch 9
21 Dec 17:57    INFO  Loading model structure and parameters from saved/M2SR-Dec-21-2023_15-05-18.pth
21 Dec 17:57    INFO  best valid result: OrderedDict([('recall@10', 0.1007), ('mrr@10', 0.0345), ('ndcg@10', 0.0499), ('
hit@10', 0.1007), ('precision@10', 0.0101)])
21 Dec 17:57    INFO  test result: OrderedDict([('recall@10', 0.0901), ('mrr@10', 0.0256), ('ndcg@10', 0.0404), ('hit@10
', 0.0901), ('precision@10', 0.009)])
(recbole38) root@LAPTOP-PQ18V36N:~/M2SR# █
```

Figure 5: Test result in ML-1M

less than the maximumsequence length, and use the most recent interactions from userswith more interactions than the maximum sequence length. We only used GELU as the nonlinear activation across all mod-els for fair comparison. To achieve efficient text modeling, we incorporate the pre-trained bert-base-uncased provided by hug-gingface for text data preprocessing. The implementation code is available online to ease reproducibility.We stop training after 200 epochs.Figure 4 shows the training details and the test results in ML-100k.And Figure 5 shows the training details and the test results in ML-1M.

Conclusion

This paper proposes M2SR, an MLP-based architecture for multi-modal sequential recommendations. Specifically, we design a unique Feature Mixer Layer to simultaneously extract image, text, and item sequence information. We also have a Fusion Mixer Layer to fuse these representations, and a Prediction Layer to generate recommendations. Compared to existing approaches, M2SR shows superior capabilities in extracting and fusing multi-modal data, while preserving linear computational complexity. Extensive experiments on two benchmark datasets prove that M2SR consistently surpasses other baseline methods. As a pioneering work in multi-modal sequential recommendation context, M2SR demonstrates high efficacy in combining multi-modal information. Furthermore, compatibility analysis shows our proposed mechanism of using multi-modal data can enhance other existing methods.

References

Cheng, H.-T.; Koc, L.; Harmsen, J.; Shaked, T.; Chandra, T.; Aradhye, H.; Anderson, G.; Corrado, G.; Chai, W.; Ispir, M.; et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, 7–10.

Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Covington, P.; Adams, J.; and Sargin, E. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*, 191–198.

Deldjoo, Y.; Elahi, M.; Cremonesi, P.; Garzotto, F.; Piazzolla, P.; and Quadrana, M. 2016. Content-based video recommendation system based on stylistic visual features. *Journal on Data Semantics*, 5: 99–113.

Fusco, F.; Pascual, D.; and Staar, P. 2022. pNLP-mixer: an efficient all-MLP architecture for language. *arXiv preprint arXiv:2202.04350*.

Guo, H.; Tang, R.; Ye, Y.; Li, Z.; and He, X. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247*.

Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; and Tikk, D. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.

Hou, Y.; Mu, S.; Zhao, W. X.; Li, Y.; Ding, B.; and Wen, J.-R. 2022. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 585–593.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, 197–206. IEEE.

Koren, Y.; Bell, R.; and Volinsky, C. 2009. Matrix factorization techniques for recommender systems. *Computer*, 42(8): 30–37.

Li, M.; Zhao, X.; Lyu, C.; Zhao, M.; Wu, R.; and Guo, R. 2022. MLP4Rec: A pure MLP architecture for sequential recommendations. *arXiv preprint arXiv:2204.11510*.

Liu, Q.; Wu, S.; Wang, D.; Li, Z.; and Wang, L. 2016. Context-aware sequential recommendation. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 1053–1058. IEEE.

McAuley, J.; Targett, C.; Shi, Q.; and Van Den Hengel, A. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, 43–52.

Shan, Y.; Hoens, T. R.; Jiao, J.; Wang, H.; Yu, D.; and Mao, J. 2016. Deep crossing: Web-scale modeling without manually crafted combinatorial features. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 255–262.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 1441–1450.

Tolstikhin, I. O.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Steiner, A.; Keysers, D.; Uszkoreit, J.; et al. 2021. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34: 24261–24272.

Touvron, H.; Bojanowski, P.; Caron, M.; Cord, M.; El-Nouby, A.; Grave, E.; Izacard, G.; Joulin, A.; Synnaeve, G.; Verbeek, J.; et al. 2022. Resmlp: Feedforward networks for image classification with data-efficient training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 5314–5321.

Van den Oord, A.; Dieleman, S.; and Schrauwen, B. 2013. Deep content-based music recommendation. *Advances in neural information processing systems*, 26.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wu, C.; Wu, F.; Qi, T.; and Huang, Y. 2021. Mmrec: multimodal news recommendation. *arXiv preprint arXiv:2104.07407*.

Wu, F.; Qiao, Y.; Chen, J.-H.; Wu, C.; Qi, T.; Lian, J.; Liu, D.; Xie, X.; Gao, J.; Wu, W.; et al. 2020. Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 3597–3606.

Wu, S.; Tang, Y.; Zhu, Y.; Wang, L.; Xie, X.; and Tan, T. 2019. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 346–353.

Yuan, F.; Zhang, G.; Karatzoglou, A.; Jose, J.; Kong, B.; and Li, Y. 2021. One person, one model, one world: Learning continual user representation without forgetting. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 696–705.

Yuan, G.; Yuan, F.; Li, Y.; Kong, B.; Li, S.; Chen, L.; Yang, M.; Yu, C.; Hu, B.; Li, Z.; et al. 2022. Tenrec: A Large-scale Multipurpose Benchmark Dataset for Recommender Systems. *Advances in Neural Information Processing Systems*, 35: 11480–11493.

Zhang, T.; Zhao, P.; Liu, Y.; Sheng, V. S.; Xu, J.; Wang, D.; Liu, G.; Zhou, X.; et al. 2019. Feature-level Deeper Self-Attention Network for Sequential Recommendation. In *IJCAI*, 4320–4326.

Zhou, K.; Yu, H.; Zhao, W. X.; and Wen, J.-R. 2022. Filter-enhanced MLP is all you need for sequential recommendation. In *Proceedings of the ACM web conference 2022*, 2388–2399.