

C-TLSAN: Content-Enhanced Time-Aware Long- and Short-Term Attention Network for Personalized Recommendation

Siqi Liang*
lsq950917@gmail.com
Purdue University
West Lafayette, Indiana, USA

Yudi Zhang*
yudiz@iastate.edu
Iowa State University
Ames, Iowa, USA

Yubo Wang
wangyubo9398@gmail.com
Purdue University
West Lafayette, Indiana, USA

Abstract

Sequential recommender systems aim to model users' evolving preferences by capturing patterns in their historical interactions. Recent advances in this area have leveraged deep neural networks and attention mechanisms to effectively represent sequential behaviors and time-sensitive interests. In this work, we propose C-TLSAN (Content-Enhanced Time-Aware Long- and Short-Term Attention Network), an extension of the TLSAN architecture that jointly models long- and short-term user preferences while incorporating semantic content associated with items—such as product descriptions.

C-TLSAN enriches the recommendation pipeline by embedding textual content linked to users' historical interactions directly into both long-term and short-term attention layers. This allows the model to learn from both behavioral patterns and rich item content, enhancing user and item representations across temporal dimensions. By fusing sequential signals with textual semantics, our approach improves the expressiveness and personalization capacity of recommendation systems.

We conduct extensive experiments on large-scale Amazon datasets, benchmarking C-TLSAN against state-of-the-art baselines, including recent sequential recommenders based on Large Language Models (LLMs), which represent interaction history and predictions in text form. Empirical results demonstrate that C-TLSAN consistently outperforms strong baselines in next-item prediction tasks. Notably, it improves AUC by 1.66%, Recall@10 by 93.99%, and Precision@10 by 94.80% on average over the best-performing baseline (TLSAN) across 10 Amazon product categories. These results highlight the value of integrating content-aware enhancements into temporal modeling frameworks for sequential recommendation. Our code is available at <https://github.com/booml247/cTLSAN>.

CCS Concepts

• Information systems → Recommender systems.

*Both authors contributed equally to this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GenAIRecP@KDD 2025, Toronto, CA

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
<https://doi.org/XXXXXXX.XXXXXXX>

Keywords

Content-aware Sequential Recommendation, Time-aware Attention Mechanism, Personalized Recommendation, User Behavior Modeling, Next-item Prediction

ACM Reference Format:

Siqi Liang, Yudi Zhang, and Yubo Wang. 2025. C-TLSAN: Content-Enhanced Time-Aware Long- and Short-Term Attention Network for Personalized Recommendation. In *Proceedings of In Proceedings of Second Workshop on Generative AI for Recommender Systems and Personalization (GenAIRecP@KDD 2025)*. ACM, Toronto, Canada, 6 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

With the growing demand for personalized user experiences, delivering timely and semantically rich recommendations has become a cornerstone of digital marketing. Traditional static models are being replaced by methods that capture the dynamic nature of user behavior[10]. Sequential recommendation reflects this shift by modeling interactions as ordered sequences, uncovering temporal dependencies and behavioral patterns [4, 5]. Deep learning has accelerated progress in this area, enabling models to learn complex user-item interactions and time-sensitive patterns [13]. These methods have gained traction across domains such as e-commerce and streaming, where understanding the temporal flow of user interests enhances engagement and conversion [2, 11].

Despite these advances, real-world deployment of sequential recommenders remains challenging. Practical systems must model both short- and long-term dependencies while incorporating diverse signals—such as user histories, item metadata (e.g., price, brand), and textual content (e.g., titles, descriptions). They must also scale efficiently, adapt to user changes, and support rapid experimentation.

Recently, Large Language Models (LLMs) have been applied to sequential recommendation by framing the task as language modeling—encoding user histories and predicting items using text prompts [8, 9]. Though promising, these methods often struggle to capture fine-grained temporal patterns and structured behavioral signals.

To address these limitations, we propose C-TLSAN (Content-Aware Time-aware Long- and Short-term Attention Network), an extension of TLSAN [16], which integrates item-level textual content into temporal modeling. C-TLSAN enhances user and item representations by incorporating product descriptions and metadata into both long- and short-term attention layers, enabling more robust and personalized recommendations. The main contributions of this work include:

- We introduce C-TLSAN, a content-aware sequential recommendation model that unifies temporal attention with item-level textual content, improving the quality of user-item representations.
- We benchmark C-TLSAN against state-of-the-art LLM-based recommendation methods, showing that our model consistently outperforms them in accuracy and robustness. We also highlight the limitations of LLM-based approaches in capturing structured and temporal patterns.
- Extensive experiments on ten large-scale Amazon datasets demonstrate that C-TLSAN delivers significant improvements over competitive baselines across multiple metrics.

2 Related Work

Sequential recommendation has attracted substantial research attention in recent years [1, 6, 15, 16, 18]. SHAN [15] leverages an attention mechanism to model both long- and short-term user preferences, incorporating context through user embeddings. However, it overlooks the effect of time decay in long-term behavior sequences and fails to capture fine-grained user-item interactions across different feature dimensions. Building on these limitations, Zhang et al. [16] proposed TLSAN, which introduces a dual-attention mechanism to independently model long-term and short-term user behaviors while explicitly incorporating temporal dynamics. Inspired by the Deep Interest Network (DIN) [19], TLSAN effectively balances recent user interests with historically persistent preferences by applying personalized, time-aware attention mechanisms to both temporal scopes.

While transformer-based architectures dominate sequential recommendation, Fan et al. [3] introduced TiM4Rec, which employs a time-aware structured state space model. This novel approach offers an alternative to attention mechanisms by utilizing state space modeling to capture temporal dynamics and sequential dependencies in user behaviors.

The integration of LLMs in recommendation systems has emerged as a promising direction. Yang et al. [14] proposed LRD (Sequential Recommendation with Latent Relations based on LLM), which leverages LLMs to discover and utilize implicit relationships between products, enriching the recommendation model with semantic understanding. Qu et al. [9] presented a critical examination of pre-trained language models in sequential recommendation, demonstrating how LLM embeddings can be effectively used to initialize ID embeddings in traditional deep learning recommendation models. Their work bridges the gap between semantic-rich LLM representations and conventional recommendation architectures. Taking a more radical approach, Li et al. [8] proposed CALRec, a purely LLM-based recommendation framework that utilizes contrastive alignment of generative LLMs for sequential recommendation tasks.

3 Method

3.1 Problem Formulation

Let $\mathcal{U} = \{u_1, u_2, \dots, u_M\}$, $\mathcal{I} = \{j_1, j_2, \dots, j_N\}$ and $\mathcal{C} = \{c_1, c_2, \dots, c_K\}$ denote the sets of users, items and item categories, respectively. For each user $u \in \mathcal{U}$, we define their interaction history up to time t as $\mathcal{L}_t^u = \{S_1^u, S_2^u, \dots, S_t^u\}$, where each $S_i^u \subseteq \mathcal{I}$ ($i \in [1, t]$) represents a session containing items the user interacted with at time i .

Following the architecture of TLSAN, we divide the interaction history into two parts: the long-term behavior sequence $\mathcal{L}_{t-1}^u$, capturing user preferences across past sessions, and the short-term behavior sequence S_t^u , representing recent session-level interactions, where t is the current time.

The goal is to accurately predict the next item a user is likely to interact with, based on their historical behavior, and recommend the most relevant items accordingly.

3.2 Fusing Content Information

Baseline models typically represent users, items, and categories using discrete ID embeddings. While effective in capturing collaborative signals, such approaches overlook valuable item-level content information—such as product descriptions and metadata—which can provide richer semantic cues about item characteristics and user intent.

To address this limitation, we enhance the model by integrating content-based item representations. Specifically, we define the item content set as $\mathcal{M} = m_1, m_2, \dots, m_N$, where m_i corresponds to the content features (e.g., title, description) of item $i \in \mathcal{I}$. These content features are embedded into dense semantic vectors and fused with traditional ID-based embeddings, enabling the model to capture more nuanced item semantics and improve its ability to generate relevant recommendations.

The remaining architecture follows TLSAN [16]. During training, we use all user interaction histories in the training set to learn user-item interaction patterns. The model operates in a binary classification setting, predicting whether a user will engage with a candidate item in the test set. Figure 1 illustrates the architecture of C-TLSAN. User behavior histories are segmented into long-term and short-term sessions based on timestamps, and a random item from the short-term session is selected as the prediction target. The model comprises three main components:

(a) Dynamic User Category Extraction: From the long-term behavior sequence and category embeddings, a dynamic user category is derived to obtain a user-specific category representation $u_{e,t}$.

(b) Long-Term Preference Extraction: For the long-term session, item ID and category embeddings are concatenated with item content embeddings to form the long-term behavior representation $\mathcal{L}t - 1^u$. Personalized time position embeddings $\mathcal{P}t - 1$ are added to model user-specific temporal dynamics, yielding temporally and semantically enriched historical embeddings $\mathcal{H}t - 1$. Feature-wise attention is applied to extract the user’s long-term preferences, denoted as $ut - 1$.

(c) Short-Term Preference Extraction: Similarly, item ID and category embeddings are combined with content embeddings for the short-term session to produce S_t^u . This is concatenated with $ut - 1$, and feature-wise attention is applied to compute the unified user representation u_t .

For final prediction, the user category representation $u_{e,t}$ and short-term preference u_t are summed to form a comprehensive user vector. A sigmoid cross-entropy loss is used, where the positive sample is the ground-truth next item and negative samples are randomly drawn. This loss function encourages the model to prioritize relevant items while suppressing irrelevant ones.

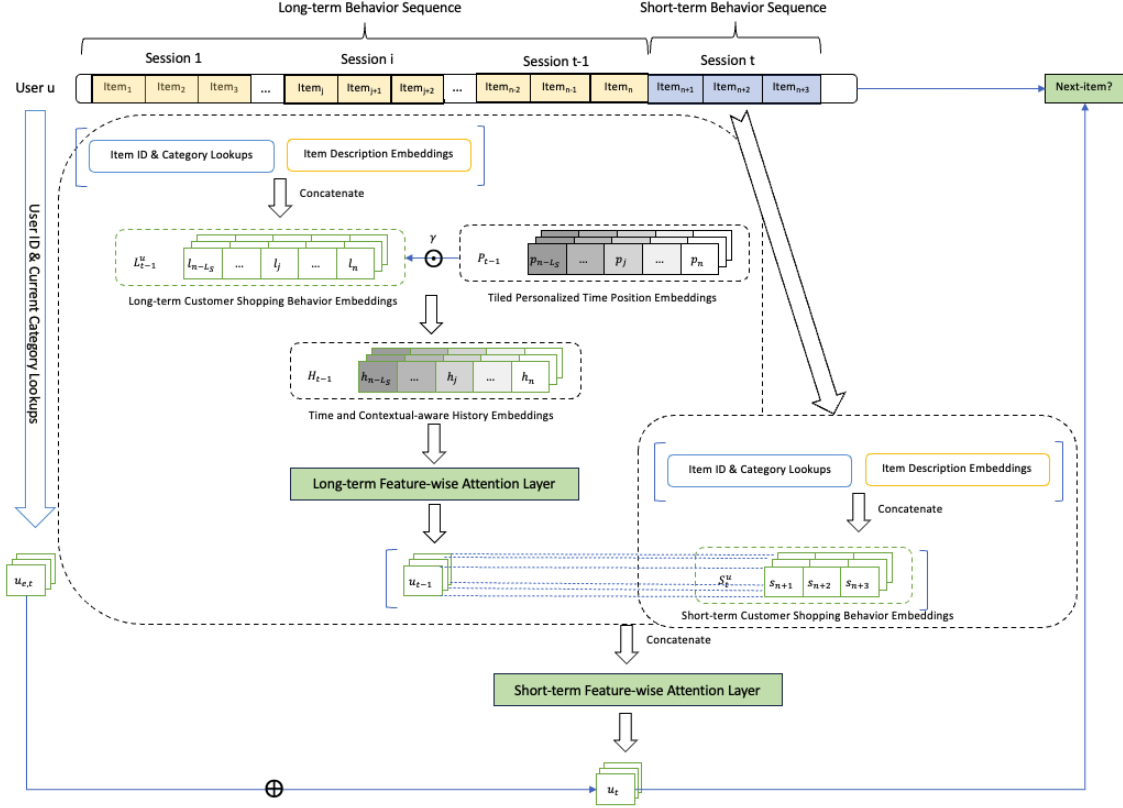


Figure 1: Model architecture of C-TLSAN.

3.3 LLM Based Sequential Recommendation

You are a smart recommendation assistant.
Here is the customer's recent browsing history:

1. [Product description 1]
2. [Product description 2]
3. [Product description 3]
- ...

Now, we want to recommend **one** of the following two items:

[Product ID 1]: [Product description]
[Product ID 2]: [Product description]

Based on the customer's interest, which item would you recommend? Respond in the format:
Recommendation: [Product ID 1 or Product ID 2]
Reason: [your explanation]

Figure 2: Example prompt for sequential product recommendation task.

Inspired by recent advances in sequential recommendation using LLMs [17], we explore a simplified yet effective approach that leverages prompt engineering to instruct LLMs in next-item prediction tasks. The core idea is to reframe the sequential recommendation

problem as a natural language understanding task, where a user's interaction history and candidate items are expressed in text form and passed as input to a pretrained LLM (LLAMA 3.3 70B [12]).

Figure 2 illustrates an example prompt template used in our experiments. The prompt presents the model with a user's browsing history encoded as a list of product descriptions, followed by two candidate items. The LLM is then asked to choose the most appropriate next item and explain its reasoning in natural language.

In our experiments, we compare this prompt-based LLM recommendation approach against C-TLSAN and other neural baselines. The results demonstrate that while LLMs exhibit promising performance in some scenarios, C-TLSAN still outperforms them consistently across most datasets, highlighting the benefits of dedicated temporal modeling and the integration of rich item content information.

4 Results

4.1 Metrics Comparison

To comprehensively evaluate the effectiveness of C-TLSAN across various domains, we conduct experiments on the publicly available datasets from Amazon¹, a large-scale collection of user-item interaction records across a wide range of product categories. The

¹<http://jmcauley.ucsd.edu/data/amazon/>

Table 1: AUC on the public Amazon product datasets. Bold font and underlined value indicate the optimal result and the suboptimal result, separately.

Datasets	LLM Recommender	CSAN	ATRank	Bi-LSTM	PACA	TLSAN	C-TLSAN
CDs and Vinyl	0.827	0.813	0.889	0.881	0.801	0.942	0.938
Clothing Shoes and Jewelry	0.639	0.761	0.663	0.668	0.798	<u>0.927</u>	0.938
Digital-Music	0.808	0.729	0.825	0.792	0.963	<u>0.972</u>	0.974
Office Products	0.516	0.824	0.921	0.856	0.910	<u>0.969</u>	0.976
Movies and TV	0.813	0.797	0.860	0.824	0.806	<u>0.879</u>	0.909
Beauty	0.618	0.727	0.806	0.773	0.859	<u>0.925</u>	0.947
Home and Kitchen	0.592	0.702	0.736	0.684	0.788	<u>0.865</u>	0.895
Video Games	0.587	0.807	0.870	0.820	<u>0.917</u>	0.914	0.933
Toys and Games	0.678	0.812	0.829	0.775	0.861	<u>0.922</u>	0.936
Electronics	0.587	0.811	0.841	0.811	0.835	<u>0.894</u>	0.913

dataset contains detailed information on users, items, reviews, and product metadata, making it a widely adopted benchmark for recommendation research. The datasets are preprocessed following two steps:

- Filtering: Users with fewer than 10 interactions and items with fewer than 8 interactions are removed to ensure data quality.
- Behavior Range: Users with between 5 and 90 transactions are retained to capture both long-term and short-term behavior patterns within a recent three-month period.

User behavior records are segmented into ordered sessions based on daily interactions. For training, the target (next) item is randomly selected from the newest session at time step t . If the session contains only one item, the first item from the next session (time $t+1$) is chosen instead. For models explicitly handling long- and short-term user behaviors (e.g., TLSAN[16]): The newest session (excluding the target item) is treated as the short-term context, while earlier sessions (1 to $t-1$) form the long-term history. For other baseline models (ATRank[18], Bi-LSTM[7], PACA[1], CSAN[6], LLM), all sessions before time t are used as historical interactions. The item following the newest session is selected as the test item to evaluate the models. The pre-trained embedding model used in C-TLSAN is *all-MiniLM-L6-v2*. To ensure the fairness and comparability of the experiments, we keep the common parameters in each model the same and the unique parameters the optimal. All the experiments in this paper are implemented with Python 3.11.11 and Tensorflow 2.17.0, and run on a server with 192 vCPUs (AMD EPYC 2nd Gen) and 8 × NVIDIA A10G (192 GB total GPU memory).

Table 1, 2 and Figure 3 summarize the performance comparisons. The proposed C-TLSAN is the best performer while its predecessor TLSAN performs the second best. Across all datasets, cTLSAN either matches or surpasses TLSAN, demonstrating the added value of integrating textual item information. On average, cTLSAN improves over TLSAN by 1.66% in AUC, with notable gains observed in categories such as Office Products (from 0.969 to 0.976), Beauty (from 0.925 to 0.947), and Home and Kitchen (from 0.865 to 0.895). Besides, it enhances Recall@10 by 93.99% and Precision@10 by 94.80% on average over the best-performing baseline TLSAN across Amazon product datasets. These results validate the effectiveness of fusing content semantics with time-aware attention for capturing richer user-item relationships.

Table 2: Recall@10 and Precision@10 on the public Amazon product datasets.

Dataset	Recall @10		Precision @10	
	TLSAN	cTLSAN	TLSAN	cTLSAN
CDs_and_Vinyl	4.12%	8.57%	0.41%	0.86%
Digital-Music	16.38%	24.87%	1.64%	2.49%
Office_Products	31.25%	46.09%	3.13%	4.61%
Movies_and_TV_5	1.64%	6.04%	0.16%	0.60%
Beauty	11.27%	18.54%	1.13%	1.85%
Home_and_Kitchen	7.27%	12.90%	0.73%	1.29%
Video_Games	7.88%	12.28%	0.79%	1.23%
Toys_and_Games	10.53%	20.24%	1.05%	2.02%
Electronics	4.72%	8.51%	0.47%	0.85%

In light of the recent surge in interest surrounding LLMs, we also compare C-TLSAN with a prompt-engineered LLM-based recommender. While the LLM recommender demonstrates competitive performance in a few domains—such as CDs and Vinyl and Movies and TV—it consistently underperforms compared to C-TLSAN across all evaluated datasets. The average AUC gap is considerable, underscoring the limitations of prompt-based LLMs in modeling fine-grained temporal dependencies and structured behavioral patterns. For instance, in the Office Products category, C-TLSAN achieves an AUC of 0.976, significantly outperforming the LLM-based method, which scores just 0.516. These results highlight the strengths of specialized architectures like C-TLSAN, which are explicitly designed to integrate both textual semantics and time-sensitive behavioral signals in sequential recommendation scenarios. Methodologically, the performance gap stems from several key limitations of the LLM-based approach:

- LLMs treat user behavior history as plain text, with no inherent understanding of time intervals, session boundaries, or recency bias—factors that are central to effective sequential recommendation.
- While LLMs excel in capturing semantic relationships from free text, they struggle to model structured signals like category hierarchies, frequency patterns, and multi-session dependencies without additional scaffolding or feature engineering.

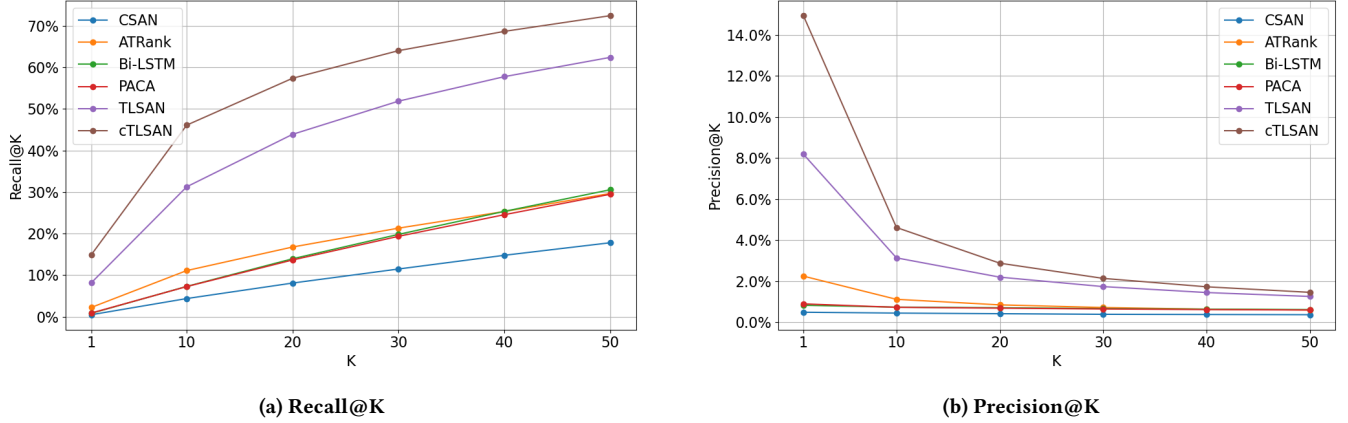


Figure 3: Recall@K and Precision@K on Amazon Office Products dataset.

- The effectiveness of LLM-based recommendation is highly dependent on prompt design. Small changes in prompt format or phrasing can lead to significant performance variation, making the model harder to optimize and less reliable in production settings.
- LLMs require large-scale inference resources and can be prohibitively expensive for real-time applications, whereas C-TLSAN is comparatively lightweight and designed for scalable deployment.

In contrast, cTLSAN addresses these challenges directly. By incorporating both long- and short-term user behavior through personalized attention mechanisms and fusing textual information (e.g., item descriptions) into the representation space, it captures both semantic and sequential dynamics in a unified and interpretable framework.

4.2 Limitations of LLMs in Content-Aware Recommendation

A key limitation of using LLMs for recommendation is their tendency to rely on surface-level lexical similarity rather than deeper user intent or behavioral context.

Consider a user whose browsing history includes: (1) Avery 5267 bright white laser printer labels, (2) Pilot B2P recycled pens, (3) GP Ink Jet and Laser Paper, (4) a dual-tip eraser, and (5) Pilot’s Metropolitan Collection pens. This sequence reflects a clear behavioral pattern centered on printing, writing tools, and workspace optimization. However, the LLM recommended a **Post-it Super Sticky Color-Coding Label** instead of a more functionally relevant product, such as a **desktop printer stand with casters**. This misstep was driven by overemphasis on the keyword “labels” rather than an understanding of the user’s broader task-oriented goal: optimizing a high-usage printing environment.

A similar issue arose in a scenario where the user’s activity involved packing and organization tasks—browsing items such as a desktop file case, wardrobe moving boxes, and heavy-duty sealing tape. Despite this consistent behavioral context, the LLM recommended **binder dividers** instead of a more practical choice

like **non-toxic, multicolor chalk**, which could be used for labeling or categorizing during the packing process. The model likely over-weighted weak textual associations (e.g., between “tabs” and “folders”) while missing the task-relevant semantics underlying the user’s actions.

In both cases, a deep learning model trained on behavioral sequences—such as co-clicks, purchases, and session-level intent—could more effectively infer the user’s underlying goal and produce semantically coherent recommendations. This highlights the importance of leveraging structured behavioral context and temporal signals alongside textual understanding in recommendation systems.

5 Conclusion

In this paper, we introduced C-TLSAN, a content-aware, time-sensitive sequential recommendation model designed to capture both long- and short-term user preferences while integrating rich semantic item information such as product descriptions and metadata. By enhancing traditional ID-based embeddings with item-level content features and modeling user behavior through temporal attention mechanisms, C-TLSAN provides a more nuanced understanding of user intent.

Through comprehensive experiments on ten large-scale Amazon datasets, we demonstrated that C-TLSAN consistently surpasses state-of-the-art baselines, including recent approaches based on Large Language Models (LLMs). While LLMs offer impressive generalization and zero-shot capabilities, our findings highlight their limitations in capturing fine-grained temporal patterns and structured behavioral context—factors that are essential for high-quality personalized recommendations.

Looking forward, we see promising research directions in developing hybrid models that combine the generative reasoning and flexibility of LLMs with the structure-aware, behavior-driven strengths of deep recommendation networks. Such models could dynamically adapt to user context, incorporate multimodal content (e.g., images, user reviews), and enable more explainable and controllable recommendations.

References

- [1] Yi Cao, Weifeng Zhang, Bo Song, Weike Pan, and Congfu Xu. Position-aware context attention for session-based recommendation. *Neurocomput.*, 376(C): 65–72, February 2020. ISSN 0925-2312. doi: 10.1016/j.neucom.2019.09.016. URL <https://doi.org/10.1016/j.neucom.2019.09.016>.
- [2] Xu Chen, Hongteng Xu, Yongfeng Zhang, Jiaxi Tang, Yixin Cao, Zheng Qin, and Hongyuan Zha. Sequential recommendation with user memory networks. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM)*, pages 108–116, 2018. doi: 10.1145/3159652.3159668.
- [3] Hao Fan, Mengyi Zhu, Yanrong Hu, Hailin Feng, Zhijie He, Hongjiu Liu, and Qingyang Liu. Tim4rec: An efficient sequential recommendation model based on time-aware structured state space duality model, 2024. URL <https://arxiv.org/abs/2409.16182>.
- [4] Hui Fang, Danning Zhang, Yiheng Shu, and Guibing Guo. Deep learning for sequential recommendation: Algorithms, influential factors, and evaluations. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 1652–1655, 2020. doi: 10.1145/3397271.3401131.
- [5] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *International Conference on Learning Representations (ICLR)*, 2016. URL <https://arxiv.org/abs/1511.06939>.
- [6] Xiaowen Huang, Shengsheng Qian, Quan Fang, Jitao Sang, and Changsheng Xu. Csan: Contextual self-attention network for user sequential recommendation. In *Proceedings of the 26th ACM International Conference on Multimedia, MM '18*, page 447–455, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356657. doi: 10.1145/3240508.3240609. URL <https://doi.org/10.1145/3240508.3240609>.
- [7] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *International conference on machine learning*, pages 2342–2350. PMLR, 2015.
- [8] Yaoyiran Li, Xiang Zhai, Moustafa Alzantot, Keyi Yu, Ivan Vulić, Anna Korhonen, and Mohamed Hammad. Calrec: Contrastive alignment of generative llms for sequential recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys '24*, page 422–432. Association for Computing Machinery, 2024. ISBN 9798400705052. doi: 10.1145/3640457.3688121. URL <https://doi.org/10.1145/3640457.3688121>.
- [9] Zekai Qu, Ruobing Xie, Chaojun Xiao, Zhanhui Kang, and Xingwu Sun. The elephant in the room: Rethinking the usage of pre-trained language model in sequential recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys '24*, page 53–62, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400705052. doi: 10.1145/3640457.3688107. URL <https://doi.org/10.1145/3640457.3688107>.
- [10] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. Sequence-aware recommender systems. *ACM Computing Surveys*, 51(4):66:1–66:36, 2018. doi: 10.1145/3190616.
- [11] 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)*, pages 1441–1450, 2019. doi: 10.1145/3357384.3357895.
- [12] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, et al. Llama 3 technical report, 2024. arXiv preprint arXiv:2404.07143.
- [13] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z. Sheng, Mehmet A. Orgun, and Defu Lian. A survey on session-based recommender systems. *ACM Computing Surveys*, 54(7):1–38, 2019. doi: 10.1145/3465401.
- [14] Shenghao Yang, Weizhi Ma, Peijie Sun, Qingyao Ai, Yiqun Liu, Mingchen Cai, and Min Zhang. Sequential recommendation with latent relations based on large language model, 2024. URL <https://arxiv.org/abs/2403.18348>.
- [15] Haochao Ying, Fuzhen Zhuang, Fuzheng Zhang, Yanchi Liu, Guandong Xu, Xing Xie, Hui Xiong, and Jian Wu. Sequential recommender system based on hierarchical attention network. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18*, page 3926–3932. AAAI Press, 2018. ISBN 9780999241127.
- [16] Jianqing Zhang, Dongjing Wang, and Dongjin Yu. Tlsan: Time-aware long- and short-term attention network for next-item recommendation. *Neurocomputing*, 441:179–191, 2021. ISSN 0925-2312. doi: <https://doi.org/10.1016/j.neucom.2021.02.015>. URL <https://www.sciencedirect.com/science/article/pii/S0925231221002605>.
- [17] Zhi Zheng, WenShuo Chao, Zhaopeng Qiu, Hengshu Zhu, and Hui Xiong. Harnessing large language models for text-rich sequential recommendation. In *Proceedings of the ACM Web Conference 2024, WWW '24*, page 3207–3216, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400701719. doi: 10.1145/3589334.3645358. URL <https://doi.org/10.1145/3589334.3645358>.
- [18] Chang Zhou, Jinze Bai, Junshuai Song, Xiaofei Liu, Zhengchao Zhao, Xiuxi Chen, and Jun Gao. Atrank: an attention-based user behavior modeling framework for recommendation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'18/AAAI'18/EAAI'18*. AAAI Press, 2018. ISBN 978-1-57735-800-8.
- [19] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junji Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1059–1068, 2018.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009