Multi-view Enhanced Graph Attention Network for Session-based Music Recommendation

DONGJING WANG, Hangzhou Dianzi University, China
XIN ZHANG, Hangzhou Dianzi University, China, Hangzhou Dianzi University Shangyu Institute of Science and Engineering, China, and Nanjing University of Science and Technology, China
YUYU YIN and DONGJIN YU, Hangzhou Dianzi University, China
GUANDONG XU, University of Technology Sydney, Australia
SHUIGUANG DENG, Zhejiang University, China

Traditional music recommender systems are mainly based on users' interactions, which limit their performance. Particularly, various kinds of content information, such as metadata and description, can be used to improve music recommendation. However, it remains to be addressed how to fully incorporate the rich auxiliary/side information and effectively deal with heterogeneity in it. In this article, we propose a **M**ulti-view **E**nhanced **G**raph **A**ttention **N**etwork (named **MEGAN**) for session-based music recommendation. MEGAN can learn informative representations (embeddings) of music pieces and users from heterogeneous information based on graph neural network and attention mechanism. Specifically, the proposed approach MEGAN first models users' listening behaviors and the textual content of music pieces with a Heterogeneous Music Graph (HMG). Then, a devised Graph Attention Network is used to learn the low-dimensional embedding of music pieces and users and by integrating various kinds of information, which is enhanced by multi-view from HMG in an adaptive and unified way. Finally, users' hybrid preferences are learned from users' listening behaviors and music pieces that satisfy users' real-time requirements are recommended. Comprehensive experiments are conducted on two real-world datasets, and the results show that MEGAN achieves better performance than baselines, including several state-of-the-art recommendation methods.

CCS Concepts: • **Information systems** → **Recommender systems**;

Additional Key Words and Phrases: Recommender systems, music recommendation, session-based, graph neural network, attention

This research was supported by the National Natural Science Foundation of China under Grant No.62202131, Natural Science Foundation of Zhejiang Province under No. LQ21F020015, the National Natural Science Foundation of China under Grants U20A20173 and 62125206, and the Key Research Project of Zhejiang Province under Grants 2022C01145 and 2020C01165.

Authors' addresses: D. Wang, Y. Yin, and D. Yu, Hangzhou Dianzi University, No. 1158, Wenze Street, Hangzhou, Zhejiang, China, 310018; emails: {Dongjing.Wang, yinyuyu, yudj}@hdu.edu.cn; X. Zhang (corresponding author), Hangzhou Dianzi University, No. 1158, Wenze Street, Hangzhou, Zhejiang, China, 310018 and Hangzhou Dianzi University Shangyu Institute of Science and Engineering, No. 77, Fuxing Rd.(W), Shaoxing, Zhejiang, China, 312399 and Nanjing University of Science and Technology, No. 200, Xiaolingwei Street, Nanjing, Jiangsu, China, 210094; email: zhangxin@hdu.edu.cn; G. Xu, University of Technology Sydney, 15 Broadway Rd, Ultimo, Sydney, NSW, Australia, 2008; email: guandong.xu@uts.edu.au; S. Deng, Zhejiang University, No. 38, Zheda Road, Hangzhou, Zhejiang, China, 310027; email: dengsg@zju.edu.cn.

1046-8188/2023/08-ART16 $15.00
https://doi.org/10.1145/3592853

ACM Reference format:
Dongjing Wang, Xin Zhang, Yuyu Yin, Dongjin Yu, Guandong Xu, and Shuiguang Deng. 2023. Multi-view Enhanced Graph Attention Network for Session-based Music Recommendation. *ACM Trans. Inf. Syst.* 42, 1, Article 16 (August 2023), 29 pages.
https://doi.org/10.1145/3592853

1 INTRODUCTION

Recently, the digital service market has been growing rapidly due to the innovation on information technology. According to the 2021 **IFPI (International Federation of Phonographic Industry)** Global Music Report,[1] the global recorded music markets have grown by 18.5%, and the streaming music market continued to grow strongly in 2021, up by 24.3%. Meanwhile, users can access massive amounts of digital music content conveniently. Specifically, both Amazon Music[2] and Apple music[3] provide over 90 million songs for users (statistics in May 2022). As a result, it becomes more challenging for people to find the music that they enjoy from massive amount of music content, which is called information overload problem.

Recommender systems [1, 9, 25] are developed to address information overload problem by helping people to find the services or content they need from massive data available via different strategies, such as collaborative filtering-based recommendation, content-based recommendation, context-aware recommendation, sequential recommendation, and so on. Especially, the algorithmic advances of recommender systems have been used in various applications, such as online music application [27, 41], group activity website [12], point-of-interest check-in service [16], to provide personalized services. However, traditional music recommender systems still suffer from limited performance, especially for applications with massive music data. Hybrid methods [23] are proposed to improve the accuracy of recommender systems by combining traditional collaborative filtering or content-based methods with auxiliary content data such as users' profile, items' attribute, and so on. However, existing hybrid methods may not fully utilize these various kinds of data in a flexible and adaptive way.

Furthermore, users' patterns of music listening behavior have also been changing gradually in recent years. For example, according to the report by **Tencent Music Entertainment (TME)**[4] in the first quarter of 2020, the COVID-19 global pandemic brought emerging opportunities in the digital era, and the mobile **MAU (monthly active user)** and paying user growth on Tencent Music remained robust, up 13% and 19% year-over-year, respectively. Generally, the popularization of smart phones as well as the rapid development of mobile internet technologies make it possible for people to enjoy music almost wherever and whenever they want, which brings more challenges to users' preference modeling and music recommendation. Fortunately, users' preferences may be inferred from their music playing behaviors recorded by the applications on the mobile devices, and the inferring process can be enhanced by the corresponding contents data, such as metadata, description, lyrics, and so on.

Besides, users have personalized tastes for music, and they may have different preferences or interest even when they are listening to the same music pieces. For example, Figure 1 shows the listening behaviors of two users, i.e., u_1 and u_2, to five music pieces (m_1-m_5), and these two users may focus on different features of music even when listen to the same music pieces. For example, u_1 is more interested in music pieces with vocal features while u_2 prefers instrumental aspects,

[1] https://globalmusicreport.ifpi.org/.
[2] https://www.amazon.com/music/unlimited.
[3] https://www.apple.com/apple-music/.
[4] http://ir.tencentmusic.com/download/TME+Transcript+1Q20.pdf.

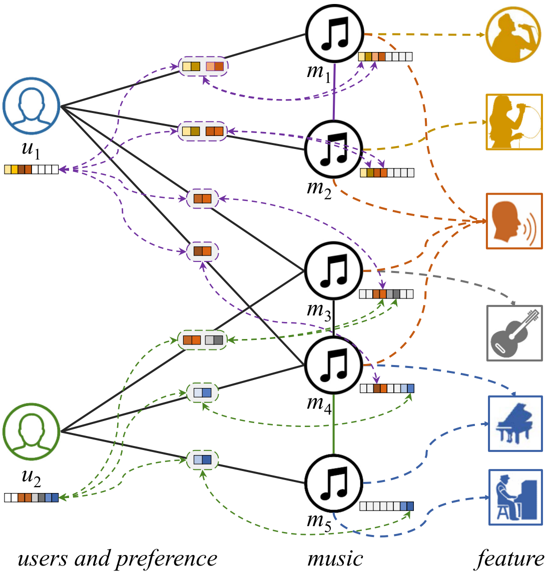


Fig. 1. Illustration of users' preferences in music listening scenario. Generally, each user has her/his personalized interest. Especially, they may focus on different features or aspects when listening to the same music pieces. For example, u_1 may be interested in the vocal features of m_3 and m_4 while u_2 prefers the instrumental aspects of those music pieces.

although both users have listened to music m_3 and m_4. Users' specific interests may reflect in the various kinds of data, including users' listening sequences and various kinds of content information. For example, the music pieces m_1 and m_2 that have been listened to by user u_1 has common genre (vocal features) with music m_3 and m_4, which may indicate u_1's preferences for vocal features. Hence, how to fully utilize those heterogeneous data is a key factor to further improve the performance of music recommendation.

Based on the analysis mentioned above, we propose a **M**ulti-view **E**nhanced **G**raph **A**ttention **N**etwork (called **MEGAN**), which can learn informative representations of users' preference and music pieces' feature from heterogeneous information based on graph neural network and attention mechanism for session-based music recommendation.

Specifically, the proposed approach MEGAN first models various kinds of data, including users' music listening behaviors and the textual content data of users and music pieces, with a **Heterogeneous Music Graph (HMG)** in a unified way. Then, a devised graph attention network, which combines graph neural network with attention mechanism, is used to embed users and music pieces into the low-dimensional real-valued space based on various kinds of information from HMG. Especially, the learning process in MEGAN is enhanced by multi-view in HMG, including users' three views (profile/behavior/session) and music pieces' four views (context/interaction/transition/attribute) in an adaptive and effective way. Note that the representation learning (including message passing and node aggregation) strategies for each view can be designed according to the specific requirements of tasks, and the attention mechanism enhances the adaptability of the learning process. Finally, we further learn users' dynamic long- and short-term preferences from their music listening records as well as the embedding of music and users and utilize them to recommend music pieces that satisfy users' real-time requirements.

Compared with other similar works, such as session-based recommendation methods, the proposed model MEGAN can: (1) exploit heterogeneous information including music listening behavior records and the content data of music pieces in a unified way, (2) learn the embedding of user and music accurately by aggregating neighbors and information from different views relevant with music recommendation tasks, (3) model and fuse users' long/short-term and dynamic preferences adaptively, all of which play important roles in accurate session-based music recommendation. The main contributions of this article are summarized as follows:

- Various kinds of data, including listening records and the content of music pieces, are encoded as **Heterogeneous Music Graph (HMG)** in a unified and adaptive way;
- We propose MEGAN model based on graph neural network and attention mechanism to learn the multi-view enhanced low-dimensional embeddings of nodes in HMG precisely and adaptively;
- We learn users' long/short-term and dynamic musical preferences and fuse them for accurate session-based music recommendation;
- Comprehensive experimental evaluations are conducted on two real-world music datasets, and the results show that the proposed method MEGAN achieves better performance than baselines, including several state-of-the-art recommendation models.

The rest of this article is organized as follows: We first review the related works in Section 2. The preliminaries and the proposed approach MEGAN are introduced in detail in Sections 3 and 4. The extensive experiments as well as the corresponding discussion are given in Section 5. We conclude our work and give the possible works to improve MEGAN in the future in Section 6.

2 RELATED WORKS

In this section, the related works are presented in three aspects, including music recommendation, attention mechanism, and graph neural network, that inspire this work.

2.1 Music Recommendation

In general, existing works on music recommendation fall into four main categories, including the content-based recommendation approaches, the **collaborative filtering (CF)** recommendation approaches, the context-aware recommendation approaches, as well as the hybrid recommendation approaches [1, 3].

Content-based recommendation methods [7, 64] mainly perform recommendation according to the content feature of music, and music pieces' content includes acoustic feature or textual metadata, such as tags, lyrics, rhythm, melody, and so on. For example, music pieces that are similar with the music pieces listened to by the users will be recommended. The content-based methods can effectively avoid the cold-start problem existing in many real-world applications. However, these recommendation methods may suffer from tedious manual works of feature engineering, which limits their wide applications.

The CF-based recommendation methods [31] are mainly divided into **user-based CF (UCF)** and **item-based CF (ICF)** methods. Specifically, as for a target user u, UCF-based methods focus on finding users (neighbors) who have similar preferences (historical listening behaviors) with u and recommend music pieces that are listened to by u's neighbors. Similarly, ICF-based approaches compute the similarity between music pieces according to the playing records and recommend music pieces that are similar with the music that user u has listened to. Compared with content-based strategies, the CF-based methods can effectively satisfy users' personalized preferences by utilizing collective intelligence. However, the performance of CF-based recommendation suffers from cold start or data sparsity problem.

Users' interests are usually influenced by their contextual situations, and context-aware recommendation modes are proposed to learn users' preferences or behavior patterns under different contexts. For example, users may listen to rock music when working out, although they prefer classical music in daily life. Specifically, the contexts used in existing recommendation methods include time [51], geographical locations [18], users' activity [22], emotional state [8], listening context [37], and so on.

Furthermore, hybrid recommendation models [23, 38] combine various kinds of recommendation methods and heterogeneous data together to address the data sparsity problem and improve the performance of recommendation. For example, Yoshii et al. [59] propose a hybrid music recommendation method to incorporate both rating and content data via a Bayesian network. Especially, the hybrid method can model unobservable user preferences as latent variables for estimating. Wang et al. [47] propose a hybrid music recommendation model, which combines both CF-based and content-based recommendation methods. La et al. [21] propose **hypergraph embeddings for music recommendation (HEMR)**, which leverages the high representative power of hypergraph data structures in combination with modern graph machine learning techniques in the context of music recommendation. In particular, the content features learned by deep belief network can improve the performance of recommendation in both the warm-start and cold-start scenarios.

Recently, some researchers began to exploit users' implicit feedbacks for improving the performance of recommendation models [10, 40]. Especially, users have much more implicit feedbacks than explicit feedbacks, especially in music recommendation tasks where users usually listen to music pieces without explicit behaviors. Moreover, sequential behaviors and the metadata in music applications attract more and more attention, and **Long Short-Term Memory (LSTM)** neural networks [54] and knowledge graph [27] are used as the recommendation strategies, which achieve improvement in music recommendation tasks. Especially, our work focuses on incorporating the data of users and music pieces in different views with a **Heterogeneous Music Graph (HMG)** and uses graph neural network and attention mechanism to further improve the performance of music recommendation.

2.2 Attention Mechanism

The attention [36], inspired by human's cognitive attention, enables the model to focus on the important parts of input data in an adaptive way, and it has been widely used in various machine learning tasks, such as map query suggestion [32], text classification [56], object detection [5], and prediction and forecast [15].

Recently, more works have begun to integrate attention mechanism in recommendation tasks. For instance, Pei et al. [28] present **Interacting Attention-gated Recurrent Network (IARN)** to learn the joint effects between user and item as well as the dynamic features for recommendation. Besides, IARN can improve the interpretability of recommendation results. Chen et al. [4] propose an **Attentive Collaborative Filtering (ACF)** model for multimedia recommendation tasks. In particular, ACF uses component- and item-level attention to capture the informative aspects of multimedia items and model users' preferences, respectively. Note that ACF model can be conveniently integrated with other classic collaborative filtering models.

Attention mechanism is also combined with other classic models for prediction or recommendation tasks. For example, **Attentional Factorization Machines (AFM)** [52] integrate **Factorization Machines (FM)** with attention mechanism to specify weight of features and their interactions for improving the performance of recommendation. Guo et al. [13] propose a **Streaming Session-based Recommendation Machine (SSRM)**, where a **Matrix Factorization (MF)**-based attention model and a reservoir-based streaming model are designed to better understand the uncertainty of user behaviors based on complex session data. Wang et al. [41] propose a content- and context-aware embedding model, which learns the dynamic feature representation of music pieces and model users' preferences via attention mechanism and convolutional neural network for accurate session-based music recommendation.

Moreover, attention model is also applied in the sequence modeling in recommendation tasks, which is similar with attention mechanism in natural language processing tasks. Ying et al. [57] design a two-layer hierarchical attention to capture the representation of users' long/short-term preferences from their historical behavior sequences for sequential recommendation. Han et al. [14] present an **adaptive deep latent factor model (ADLFM)** to model users' specific preferences for rating prediction and recommendation. Especially, ADLFM is based on the observation that users may have diverse preferences for items, which cannot be effectively modeled by traditional matrix factorization methods.

Besides, Huang et al. [16] propose a recommendation model named ATST-LSTM, which incorporates the spatial and temporal contexts in historical check-in records with LSTM and uses the attention mechanism to model the relevance in behavior sequences for accurate next point of interest recommendation. Zhang et al. [62] propose a framework called Knowledge-enhanced Session-based Recommendation with Temporal Transformer to incorporate auxiliary information when learning the item and session embeddings for improving the performance. Huang et al. [17] present a multi attention-based recommendation method to learn the vector representation of group features in different view and model group's preferences on items.

2.3 Graph Neural Network

Graph-structured data, such as social network graphs, molecular structure, and knowledge graph, are quite common in real-world applications. However, these data exist in non-Euclidean space, and the processing techniques on those data are generally different with the data in Euclidean space. As a kind of deep learning technique on graph-structured data, **graph neural network (GNN)** and its variants have been widely used in many tasks, such as fraud detection [63], graph mining [60], recommendation [55, 58], and so on.

Specifically, Ying et al. [58] develop a large-scale deep recommendation engine based on graph convolutional neural networks named PinSage. PinSage incorporates node features and graph structures to obtain the embedding of nodes with random walks. Wang et al. [44] proposed a unified approach named **Neural Graph Collaborative Filtering (NGCF)**, which learns the representation of users and music by exploiting the high-order correlations and collaborative signal for recommendation. Wang et al. [48] propose **Global Context Enhanced Graph Neural Networks (GCE-GNN)** to exploit item transitions over all sessions in a more subtle manner for better inferring the user preference of the current session and improving the performance of session-based recommendation. Qiu et al. [29] propose a **Global Attributed Graph (GAG)** neural network model with a Wasserstein reservoir, which take both the global attribute and the current session into consideration for improving the performance of streaming session-based recommendation problem. Wang et al. [45] explore intents behind a user-item interaction by using auxiliary item knowledge and propose **Knowledge Graph-based Intent Network (KGIN)** for recommendation as well as interpretable explanations.

Besides, many variants of GNN have been proposed to further improve the performance on specific tasks. For example, Li [24] propose the **Gated graph neural network (GGNN)**, which uses the **Gated Recurrent Units (GRU)** as propagation strategy to enhance the performance of GNN for bAbI artificial intelligence and graph algorithm learning. GGNN is widely adopted as an effective approach in various recommendation tasks. For instance, Wu et al. [50] propose a session-based recommendation model based on GNN, namely, SR-GNN, which can learn the embedding of item and model the complex transition in behavior sequences with GGNN. Wang et al. [39] present a point of interest recommendation model ASGNN based on GGNN and attention. Specifically, ASGNN models users' preferences and behavior patterns in their check-ins and combines users' long/short-term preferences for next POI recommendation.

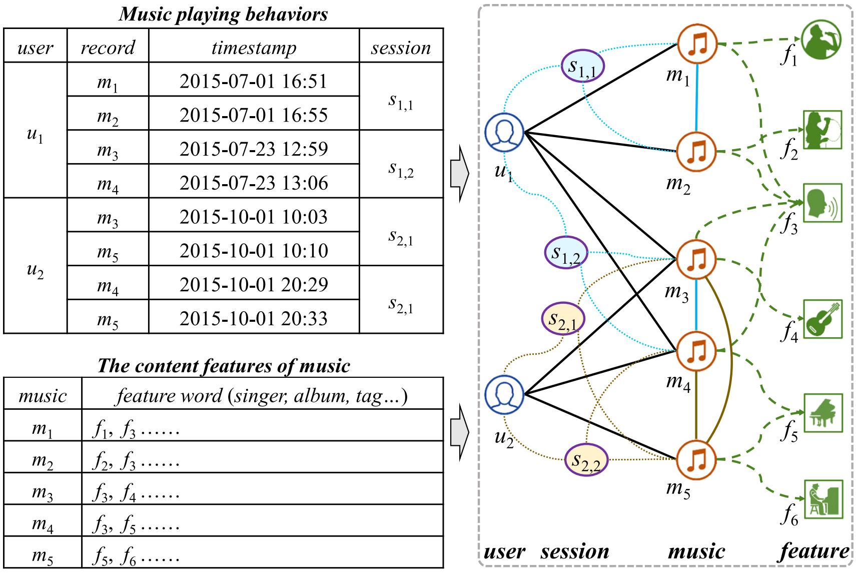
Table 1. The Key Symbols Used in This Work

Notation	**Interpretation**
$G = (V, E, W)$	the Heterogeneous Music Graph (HMG)
V, E, W	vertex set, edge set, and the set of edges' weights
$U, S, M, F \subseteq V$	user set, session set, music set and feature word set
N^x_v	the neighbors of node v in view x
$\mathbf{v}'_i \in \mathcal{R}^d$	the initial embedding of node v_i via pretraining, where d denotes the dimension
$\mathbf{v}'_f \in \mathcal{R}^{d'}$	the embedding of feature word f, where d' denotes the dimension
$\mathbf{v}^p_u, \mathbf{v}^b_u, \mathbf{v}^s_u \in \mathcal{R}^d$	user u's embedding in view of profile, behavior, and session, respectively
$\mathbf{v}^a_m, \mathbf{v}^i_m, \mathbf{v}^t_m, \mathbf{v}^c_m \in \mathcal{R}^d$	music piece m's embedding in view of attribute, interaction, transition, and context, respectively
$\mathbf{v}^{agg}$	the aggregated embedding of user and music
$\mathbf{v}_u, \mathbf{v}_m \in \mathcal{R}^d$	the final embeddings of user u's and music piece m learned by the proposed approach
$\alpha, \gamma, \beta, \varepsilon$	the attention weights
λ	gate vector in the updating process
$\mathbf{v}^{st}_u$	user u's short-term preference
$\mathbf{h}$	the hidden representation
$\Vert$	concatenation operation between vectors
$\otimes$	element-wise multiplication between vectors
$\rho^{lt}, \rho^{st}\ \rho^d$	weights of long/short-term and dynamic preference, respectively
$\mathbf{W}, \mathbf{w}, \mathbf{c}, \mathbf{q}$	the learnable matrix or vector parameters
$p_{u,m}$	the interest score of user u for music piece m

In the representation learning process of GNN, attention mechanism can be used to help specify different weights to different neighbors adaptively [46]. Graph attention networks, which combines GNN with attention networks, are proposed and applied in recommender systems. For example, Wu et al. [49] propose DiffNet++, an improved algorithm of DiffNet that models the neural influence diffusion and interest diffusion in a unified framework for social recommendation. Chen et al. [6] propose an efficient framework for session-based social recommendation. Especially, the framework can easily incorporate knowledge including social networks and capture cross-session item transitions. Song et al. [33] propose a social recommender system (DGRec) based on dynamic graph attention neural network. Especially, the proposed model DGRec can learn users' dynamic interests and model the influences between friends (users). Xu et al. [53] propose a **graph contextualized self-attention model (GC-SAN)**, which utilizes both graph neural network and self-attention mechanism to learn the local dependencies and contextualized representations for session-based recommendation. Wang et al. [43] proposed a model named KGAT to model the high-order connectivity between user-item links in the hybrid knowledge graph with graph neural network and attention mechanism for accurate top-k recommendation.

3 PRELIMINARIES

The definitions of key notations and symbols used in this article are given in Table 1. Formally, we define the user set as $U = \{u_1,\ u_2,\ u_3, \ldots, u_{|U|}\}$ and music set as $M = \{m_1,\ m_2,\ m_3, \ldots, m_{|M|}\}$, where $|U|$ and $|M|$ are the sizes of user set and music set, separately. Generally, users may listen to



Music playing behaviors

user	*record*	*timestamp*	*session*
u_1	m_1	2015-07-01 16:51	$s_{1,1}$
	m_2	2015-07-01 16:55	
	m_3	2015-07-23 12:59	$s_{1,2}$
	m_4	2015-07-23 13:06	
u_2	m_3	2015-10-01 10:03	$s_{2,1}$
	m_5	2015-10-01 10:10	
	m_4	2015-10-01 20:29	$s_{2,1}$
	m_5	2015-10-01 20:33	

The content features of music

music	*feature word* (*singer, album, tag*...)
m_1	f_1, f_3 ......
m_2	f_2, f_3 ......
m_3	f_3, f_4 ......
m_4	f_3, f_5 ......
m_5	f_5, f_6 ......

Fig. 2. Example of constructing a Heterogeneous Music Graph (HMG) from users' music playing records and the content features of music pieces

music for a period of time, and their listening behaviors are recorded as music playing sequences. Especially, each music record in sequences has corresponding timestamps, and the playing sequence is divided into different sessions based on the time interval information between records.

For example, eight music listening records of two users (u_1 and u_2) for five pieces of music (m_1-m_5) are given in Figure 2 (music listening behaviors), and they can be aggregated into four sessions based on the corresponding timestamps. Besides, the content features of music, such as singer, album, genre, tag, can help to accurately capture users' preference and music pieces' feature and improve the performance of music recommendation.

Therefore, the session-based recommendation task is defined as how to recommend appropriate music pieces that the target user may like next according to her/his playing sequence (including session) as well as music content features.

4 PROPOSED APPROACH

In this section, we present the proposed **Multi-view Enhanced Graph Attention Network (MEGAN)** in detail. As shown in Figure 3, MEGAN is composed of four main steps: (1) behavior and content data modeling with **Heterogeneous Music Graph (HMG)**; (2) **Graph Neural Network (GNN)**-based representation learning, including user interest modeling and music feature learning on HMG; (3) short-term preference inferring with attention mechanism, which captures users' short-term preferences by aggregating their recent behaviors; and (4) prediction layer, which fuses users' preference and computes the interest score of candidate music pieces for music recommendation. Next, we will introduce each part of MEGAN in detail.

4.1 Modeling with Heterogeneous Music Graph

A Heterogeneous Music Graph (HMG) is first constructed to incorporate various kinds of information, such as users' listening records and the content data of music pieces, in a unified and adaptive way. Formally, the definitions of HMG and its edges are given as follows:

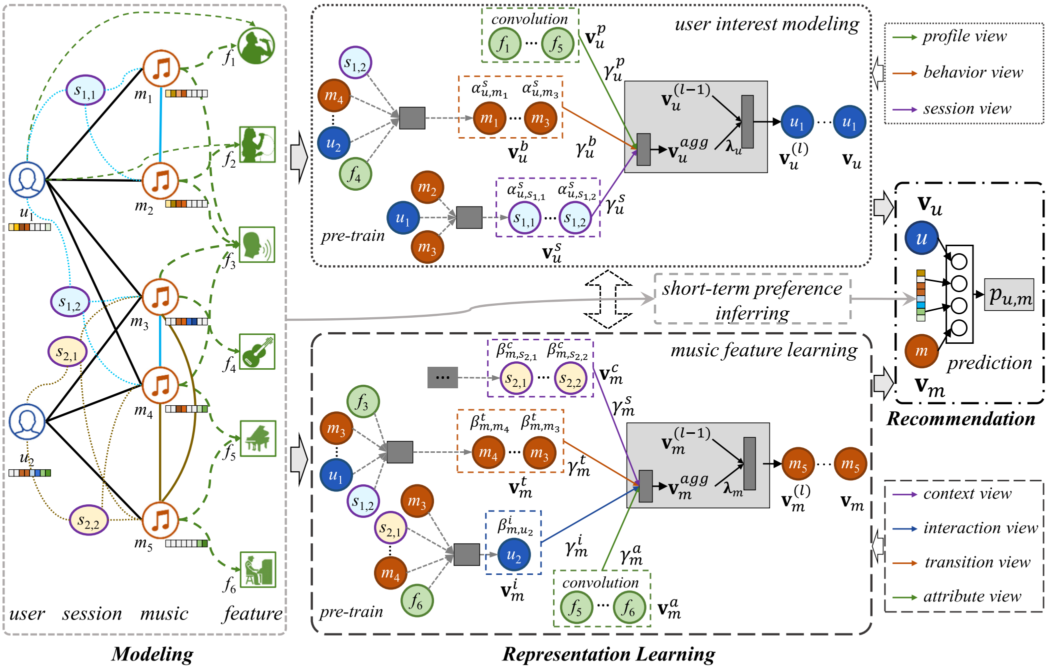


Fig. 3. The overall framework of the proposed approach MEGAN. First, Heterogeneous Music Graph (HMG) can incorporate different kinds of data and information. Then, the Graph Neural Network (GNN)-based component in MEGAN is adopted to obtain the feature representation (embedding) of user and music nodes from multi-view information on HMG. Especially, the representation learning of user node is enhanced in view of profile, behavior, and session, and the embedding of music node is learned from interaction, context, transition, and attribute views. Finally, the interest score of a user for candidate music pieces is calculated based on users' preference and the feature embedding of music.

Definition 1. The **Heterogeneous Music Graph (HMG)** is defined as a graph $G = (V, E, W)$, which has several kinds of nodes and edges. Specifically, $V = (U, S, M, F)$ represents the set of different kinds of nodes in HMG, including user node $u \in U$, session node $s \in S$, music node $m \in M$, and content feature $f \in F$, where U, S, M, and F are the user set, session set, music set, and content feature word set, separately. E is the set of different types of edges in HMG, including user-music edges, user-feature edges, user-session edges, music-feature edges, music-music edges, and music-session edge. W is the weight set of E. Formally, the weight of the edge between node v_1 and v_2 is defined as $w_{v_1,v_2} = \tanh(z) = (e^z - e^{-z})/(e^z + e^{-z})$, where z is the co-occurrence frequency of v_1 and v_2, such as how many times has user u listened to music piece m.

As shown in Figure 2, an example of constructing HMG is given. Specifically, each kind of edge represents the specific relationship between nodes. For example, there exists an edge between user u_1 and music m_1 if u_1 has listened to m_1. Besides, session nodes represent users' preferences during a specific period of time, and feature words indicate the attribute or genre of music pieces.

Note that the HMG can be extended dynamically to incorporate various kinds of data, such as music playlists or the social relationship between users. Especially, the HMG can be viewed and exploited in different aspects via its specific kind of nodes and edges (views). Besides, it is convenient to incorporate more views or design different representation learning strategies for specific views according to the requirements of tasks.

Definition 2 (Multi-view of User Node). The user nodes are connected with other nodes via user-music edges, user-feature edges, and user-session edges, which can provide important information about users' preferences. Specifically, user-music edges represent users' behavior patterns, and user-feature edges denote users' profile information, which may also indicate users' preference for music. Besides, user-session edges show users' fine-grained preferences during a specific period of time, which is quite important for session-based recommendation tasks.

In this work, three different views of music, i.e., profile, behavior, and session, can be used to enhance the representation learning of music nodes via edges between users and feature/music/session in HMG.

Definition 3 (Multi-view of Music Node). The music nodes in HMG are connected with user, music, session, and feature nodes via different edges, which can be exploited and enhanced in four different views, including interaction, context, transition, and attribute views, respectively. Note that each view of music node has its own meaning and effect. For example, the interaction view of music node helps to learn the intrinsic features of the corresponding music piece, which is similar with the idea of item-based collaborative filtering methods. Besides, the transition view can be used to capture the sequential patterns and key feature factors in music listening records.

Different views of nodes are integrated in HMG to learn robust representation of users and music pieces and help to improve the performance of music recommendation.

Definition 4 (View-aware Neighbors). Given a node v and a view x, the view-aware neighbors $N^x(v)$ are defined as the set of nodes that are connected with v via the corresponding edge type in view x in HMG. For example, as for user node u, her/his neighbors in behavior view are the music pieces that have been listened to by u.

4.2 Representation Learning via Graph Neural Network

The effective modeling of users and items is the prerequisite of accurate recommendation. Therefore, we need to learn the representation of users' preferences and the music pieces' features.

The one-hot representation/encoding is widely used in the traditional recommender systems for its ability of converting categorical data into a numerical vector. However, one-hot representation has limited capacity of modeling the intrinsic features and correlations between different users or music pieces, and it suffers from serious dimensional disaster and data sparsity problems, especially in music-related scenarios, where the size of music set can easily reach tens of millions in the real-world applications.

In this work, the proposed model MEGAN can learn the feature representation of nodes (users, music pieces, and so on) with graph neural network and attention mechanism from users' music listening behavior data as well as the content features of music pieces.

4.2.1 Embedding via Pretraining. First, we learn the pretrained embedding of the nodes in HMG with embedding model that projects node into low-dimensional feature space with the embedding matrix.

Formally, the projection process (embedding look-up) of each node $v_i \in V$, such as user node, music node, session node, feature node, and so on, is formally represented as

$$\mathbf{v}'_i = \mathbf{v}^1_i \cdot \mathbf{V}',$$

where $\mathbf{v}'_i \in \mathcal{R}^d$ is the learned embedding, $\mathbf{V}' \in \mathcal{R}^{|V|\times d}$ is node embedding matrix, d represents the dimension of nodes' embeddings, and $\mathbf{v}^1_i \in \mathcal{R}^{1\times|V|}$ is v_i's one-hot representation (encoding), where there exists a single "1" in the specific dimension and the rest dimensions are set as "0." In particular, the learned embedding can capture node features as well as their complex correlations in HMG, which is important for the effective learning of users' preferences and music pieces features as well as the recommendation tasks.

Note that the embedding matrix can be initialized randomly or via classic embedding/representation learning techniques, such as node2vec [11] and LINE [34], which can accelerate the speed of training process effectively. In this work, we adopt LINE to initialize the embedding matrix in a pre-trained way. In particular, LINE is capable of preserving both first-order and second-order proximities of networks or graphs with arbitrary types of information, which are complementary with each other. Besides, the edge-sampling algorithm in LINE can improve both the effectiveness and the efficiency of model inference process.

4.2.2 User Interest Modeling. User's interest for music is learned from three different aspects, including profile view, behavior view, and session view.

Profile view. Users' interest may reflect in their profile. We use the textual content features, such as singer, album, genre, and tags, of music pieces listened to by the user as her/his profile features, i.e., auxiliary data in profile view.

Specifically, the feature words are preprocessed based on words' **term frequency-inverse document frequency (TF-IDF)**, and only the words with high TF-IDF value will be used, since they are usually more informative and important than the ones with low TF-IDF. Then, user's interest representation (embedding) in profile view is obtained via convolution operation, which consists of three main steps: embedding looking-up, convolution, and pooling.

First, the looking-up layer transforms each feature word f_i in the profile sentences of user into corresponding word embeddings $\mathbf{v}'_{f_i} \in \mathcal{R}^{d'}$, where d' denotes the dimension of feature word embeddings. Then, each user u's profile sequence is represented with a $d' \times l'$ word embedding matrix $\mathbf{V}'_u = (\mathbf{v}'_{f_1}; \mathbf{v}'_{f_2}; \ldots; \mathbf{v}'_{f_{l'}})$, where l' is the length of the corresponding feature word sequence.

Then, the convolution layer extracts local features from word embedding matrix $\mathbf{V}'_u$ by performing convolution operation. Specifically, the numbers of input channel and output channel in convolution layer are d' and d, respectively. Besides, the size of convolving kernel is l''. Therefore, the output of the convolution layer for profile sequence of each user is an embedding matrix $\mathbf{V}''_u \in \mathcal{R}^{d \times (l'-l''+1)}$.

Finally, the hyperbolic tangent activation function and mean-pooling strategy are applied over the embedding matrix $\mathbf{V}''_u$ from the convolutional layer, and we can obtain the profile embedding of each user u as $\mathbf{v}^p_u \in \mathcal{R}^d$, where the superscript "p" indicates users' profile view.

Behavior view. Similar with the strategy of collaborative filtering-based methods, users' preferences can be learned from their historical listening behaviors, which are modeled as links (edges) between user nodes and music nodes on HMG. Therefore, the preference representation of user node can be learned from her/his music neighbors (in behavior view). Since each user may listen to various number of music pieces, we choose the k music nodes via weighted sampling strategy with replacement. Note that the u's music neighbor nodes with higher edge weights will have higher probability of being sampled.

Specifically, as for user u and her/his k music neighbors $\{m_0, m_1, \ldots, m_{k-1}\} \subseteq N^b(v_u)$, the corresponding embeddings are $\mathbf{v}'_u \in \mathcal{R}^d$ and $\mathbf{v}'_{m_0}, \mathbf{v}'_{m_1}, \ldots, \mathbf{v}'_{m_{k-1}} \in \mathcal{R}^d$. Note that these embeddings are initialized with pretrained layer. Formally, the embedding of user u in behavior view can be obtained from her/his music neighbors via attention mechanism, which is defined as:

$$\mathbf{v}^b_u = \sum_{i=0}^{k-1} \alpha_{u,m_i} \cdot \mathbf{v}'_{m_i}, \tag{1}$$

where the superscript "b" indicates the behavior view. α^b_{u,m_i} is the attention weight of user u for the ith music neighbor node m_i behavior view, which controls the influence of m_i on the representation learning of u. Formally, α_{u,m_i} is calculated based on the edge weight between u and m_i and the feature correlations between them, and it is defined with softmax operation as:

$$\alpha_{u,m_i} = softmax\left(w_{u,m_i} + a_{u,m_i}\right), \tag{2}$$

where w_{u,m_i} is the weight of the behavior edge between user u and music m_i on HMG, and it is proportional to the frequency that u has listened to m_i. Besides, a^b_{u,m_i} is the behavior score between u and m_i that indicates the correlation between users' preferences and music pieces' feature, and it is formally defined as:

$$a_{u,m_i} = \sigma\left(\mathbf{w}^\top \cdot \left(\mathbf{h}^b_u \middle\| \mathbf{h}^b_{m_i}\right)\right), \tag{3}$$

where σ is leaky ReLU activation function, and $\mathbf{w} \in \mathcal{R}^{2d''}$ is learnable vector parameter. Note that we have omitted the superscript of behavior view in $\mathbf{w}^b$ for simplicity, and the learnable parameters are independent in each view. d'' is the hidden layer size. $\|$ is concatenation operation between vectors, and $\mathbf{h}^b_u \| \mathbf{h}^b_{m_i} \in \mathcal{R}^{2d''}$. Besides, $\mathbf{h}^b_u \in \mathcal{R}^{d''}$ is the hidden representation of user u in behavior view, which is defined as the non-linear transformation of u's embedding:

$$\mathbf{h}^b_u = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_u + \mathbf{c}\right), \tag{4}$$

where $\mathbf{W} \in \mathcal{R}^{d'' \times d}$ and $\mathbf{c} \in \mathcal{R}^{d''}$ are learnable parameters. We have omitted the superscripts of behavior view in $\mathbf{W}^b$ and $\mathbf{c}^b$ for simplicity, and the learnable parameters are independent in each view. tanh () is the hyperbolic tangent activation function. Similarly, $\mathbf{h}^b_{m_i} \in \mathcal{R}^{d''}$, the hidden representation of music m_i, is defined as:

$$\mathbf{h}^b_{m_i} = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_{m_i} + \mathbf{c}\right). \tag{5}$$

Session view. Users' historical listening sequence can be divided into sessions based on the timestamps of listening records, and each session is a list of music pieces that has been listened to by a certain user within a period of time. Sessions can help to learn users' interest for music in a coarse-grained aspect, since they reflect users' specific preferences for a while.

Similar with the learning process of user embedding in behavior view, the embedding of user u in session view is the weighted sum of her/his k' session neighbor nodes, which is defined as:

$$\mathbf{v}^s_u = \sum_{i=0}^{k'-1} \alpha_{u,s_i} \cdot \mathbf{v}'_{s_i}, \tag{6}$$

where the superscript "s" indicates the session view, and $\mathbf{v}'_{s_i}$ is the embedding of u's session $s_i \subseteq N^s(v_u)$. α_{u,s_i} is the attention weight between session s_i and user u, and it controls the preferences propagation from s_i to u. Especially, the calculation process of α_{u,s_i} is similar with Equations (2)–(5), and α_{u,s_i} is formally defined as:

$$\alpha_{u,s_i} = softmax\left(w_{u,s_i} + a_{u,s_i}\right), \tag{7}$$

$$a_{u,s_i} = \sigma\left(\mathbf{w}^\top \cdot \left(\mathbf{h}^s_u \middle\| \mathbf{h}^s_{s_i}\right)\right), \tag{8}$$

$$\mathbf{h}^s_u = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_u + \mathbf{c}\right), \tag{9}$$

$$\mathbf{h}^s_{s_i} = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_{s_i} + \mathbf{c}\right), \tag{10}$$

where w_{u,s_i} is the weight of the user-session edge between user u and her/his session neighbor s_i, and its value is set as 1. a_{u,s_i} is the session score between u and s_i that indicates the correlation between users' global preferences and session preferences. σ is leaky ReLU activation function, and tanh () is the hyperbolic tangent activation function. $\mathbf{w} \in \mathcal{R}^{2d''}$, $\mathbf{W} \in \mathcal{R}^{d'' \times d}$, and $\mathbf{c} \in \mathcal{R}^{d''}$ are learnable vector/matrix parameters. $\mathbf{h}_u^s \in \mathcal{R}^{d''}$ and $\mathbf{h}_{s_i}^s \in \mathcal{R}^{d''}$ are the hidden representations of user u and session s_i.

User view aggregation and update. Then, we need to learn the embedding of user by aggregating the information and features from different views in an adaptive and effective way. Here, a gated attention model is devised for the view aggregation and update process.

Formally, the aggregated embedding of user u is defined as weighted sum of the embeddings learned in three views:

$$\mathbf{v}_u^{agg} = \gamma^p \cdot \mathbf{v}_u^p + \gamma^b \cdot \mathbf{v}_u^b + \gamma^s \cdot \mathbf{v}_u^s, \tag{11}$$

where $\mathbf{v}_u^p$, $\mathbf{v}_u^s$, and $\mathbf{v}_u^s$ are u's embeddings in profile view, behavior view, and session view, separately. γ^p, γ^b, and γ^s are the weights of three views, and they are calculated with attention mechanism. Formally, the attention weight γ^x in view x is defined with softmax function as:

$$\gamma^x = softmax\left(r^x\right), \tag{12}$$

$$r^x = \mathbf{q}^\top \cdot \tanh\left(\mathbf{W}^r \cdot \mathbf{v}_u^x + \mathbf{c}^r\right), \tag{13}$$

where $\mathbf{q} \in \mathcal{R}^{d''}$, $\mathbf{W}^r \in \mathcal{R}^{d'' \times d}$, and $\mathbf{c}^r \in \mathcal{R}^{d''}$ are learnable parameters, d'' is the size of hidden layer, and $\mathbf{v}_u^x$ is the learned embedding of user u in view x. tanh () is the hyperbolic tangent activation function.

Besides, the representation learning of nodes in the graph neural network is iterative, and the update process of user node v_u is defined with a gate mechanism as:

$$\mathbf{v}_u^{(l)} = (1 - \lambda) \otimes \mathbf{v}_u^{(l-1)} + \lambda \otimes \mathbf{v}_u^{agg}, \tag{14}$$

where $\mathbf{v}_u^{(l)} \in \mathcal{R}^d$ and $\mathbf{v}_u^{(l-1)} \in \mathcal{R}^d$ are the node representation of user u in the lth and $(l-1)$-th layer or step, respectively, and $\otimes$ is element-wise multiplication between vectors. $\lambda \in \mathcal{R}^d$ is the gate vector that controls the updating process, and it is formally defined as:

$$\lambda = \sigma_u\left(\mathbf{W}^\lambda \cdot \left(\mathbf{v}_u^{(l-1)} \middle\| \mathbf{v}_u^{agg}\right) + \mathbf{c}^\lambda\right), \tag{15}$$

where $\mathbf{W}^\lambda \in \mathcal{R}^{d \times 2d}$ and $\mathbf{c}^\lambda \in \mathcal{R}^d$ are the learnable parameters in the updating process, and σ_u is a sigmoid activation function. Especially, the subscripts of user indicators in $\mathbf{q}_u$, $\mathbf{W}_u^r$ $\mathbf{c}_u^r$, $\mathbf{W}_u^\lambda$, and $\mathbf{c}_u^\lambda$ are omitted for simplicity, and the learnable parameters are independent in the aggregation/updating process of user or music. Finally, we can get the representation (embedding) $\mathbf{v}_u$ of each user u.

4.2.3 Music Feature Modeling. A piece of music has various kinds of neighbors from which we can learn useful feature representations to effectively help behavior modeling and prediction tasks. Specifically, music's feature can be learned from four different aspects, including transition view, attribute view, context view, and interaction view.

Attribute view. Content-based music recommendation methods focus on extracting the features of music pieces from the metadata, such as songwriter, singer/artist, musical genre, acoustic, and so on, and recommend music pieces that are similar with the ones listened to by the target users. In this work, music pieces' features can be inferred from the corresponding content attribute (attribute view) to enhance the representation learning as well as recommendation tasks.

Specifically, the content words of each music piece, such as singer, album, tags, genre, and lyrics, are preprocessed based on words' **term frequency-inverse document frequency (TF-IDF)**, and only the words with high TF-IDF value will be used. Similar with the representation learning of users in the profile view, the learning process of music in attribute view is also based on convolution operation, and the attribute embedding of music m is obtained as $\mathbf{v}_m^a \in \mathcal{R}^d$, where the superscript "a" indicates music pieces' attribute view.

Interaction view. The **item-based collaborative filtering (ICF)** recommender systems perform recommendation based on the similarity between items, which is calculated using users' rating or interaction records of those items. Generally, items with common users are considered similar to each other. In this work, we enhanced the feature representation learning of music pieces by incorporating the historical interaction data (interaction view).

The learning process of music in interaction view is similar with the learning process of user in behavior view. Specifically, as for music m and its n user neighbors $\{u_0, u_2, \ldots, u_{n-1}\} \subseteq N^i(v_m)$, who have listened to m, the corresponding initialized embeddings are $\mathbf{v}'_m \in \mathcal{R}^d$ and $\mathbf{v}'_{u_0}, \mathbf{v}'_{u_1}, \ldots, \mathbf{v}'_{u_{n-1}} \in \mathcal{R}^d$. Formally, the embedding of m in interaction view can be learned from its user neighbors via attention mechanism, which is defined as:

$$\mathbf{v}_m^i = \sum_{j=0}^{n-1} \beta_{m,u_j} \cdot \mathbf{v}'_{u_j}, \tag{16}$$

where the superscript "i" represents the interaction view. β_{m,u_j} is the attention weight of the jth user neighbor for music m, and it controls the feature propagation from u_j to m. Formally, β_{m,u_j} is defined as:

$$\beta_{m,u_j} = softmax\left(w_{m,u_j} + b_{m,u_j}\right), \tag{17}$$

$$b_{m,u_j} = \sigma\left(\mathbf{w}^\top \cdot \left(\mathbf{h}_m^i \middle\| \mathbf{h}_{u_j}^i\right)\right), \tag{18}$$

$$\mathbf{h}_m = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_m + \mathbf{c}\right), \tag{19}$$

$$\mathbf{h}_{u_j} = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_{u_j} + \mathbf{c}\right), \tag{20}$$

where w_{m,u_j} is the weight of the edge between music m and user u_j on HMG, and it is proportional to the times of m being interacted with by u_j. Besides, b^i_{m,u_j} is the interaction score between m and u_j that shows the correlation between m's features and u_j's preferences. σ is leaky ReLU activation function, and tanh () is the hyperbolic tangent activation function. $\mathbf{h}_m^i \in \mathcal{R}^{d''}$ and $\mathbf{h}_{u_j}^i \in \mathcal{R}^{d''}$ are the hidden representations of music n and its user neighbor u_j on HMG. $\mathbf{w} \in \mathcal{R}^{2d''}$, $\mathbf{W} \in \mathcal{R}^{d'' \times d}$, and $\mathbf{c} \in \mathcal{R}^{d''}$ are learnable parameters. We have omitted the superscripts of interaction view in $\mathbf{w}^i$, $\mathbf{W}^i$, and $\mathbf{c}^i$ for simplicity, and the learnable parameters are independent in each view or aggregation, updating or preference inferring process.

Transition view. Generally, users may listen to music for a period of time, one piece after another, and their behaviors are recorded as listening sequences. The transition in sequences reflects users' listening patterns and also indicates the features of music pieces. For example, the music pieces that are close (or adjacent) with each other in the historical listening sequences have similar styles or features in general.

Specifically, as for music m and its n' music neighbors $\{m_0, m_1, \ldots, m_{n'-1}\} \subseteq N^t(v_m)$ (adjacent records in music listening sequences) on HMG, the embedding of m in transition view is learned from the features of $\{m_0, m_1, \ldots, m_{n'-1}\}$, and it is formally defined as follows:

$$\mathbf{v}_m^t = \sum_{j=0}^{n'-1} \beta_{m,m_j} \cdot \mathbf{v}'_{m_j}, \tag{21}$$

where $\mathbf{v}'_{m_j}$ is the embedding of m's jth music neighbor m_j, and β_{m,m_j} is the attention weight of m_j for m, and it models the correlation between m_j and m. Especially, the calculation process of attention weight β_{m,m_j} is similar with Equations (17)–(20), and β_{m,m_j} is formally defined as:

$$\beta_{m,m_j} = softmax\left(w_{m,m_j} + b_{m,m_j}\right), \tag{22}$$

$$b_{m,m_j} = \sigma\left(\mathbf{w}^\top \cdot \left(\mathbf{h}^t_m \middle\| \mathbf{h}^t_{m_j}\right)\right), \tag{23}$$

$$\mathbf{h}^t_m = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_m + \mathbf{c}\right), \tag{24}$$

$$\mathbf{h}^t_{m_j} = \tanh\left(\mathbf{W} \cdot \mathbf{v}'_{m_j} + \mathbf{c}\right), \tag{25}$$

where w_{m,m_j} is the weight of music-music transition edge, and it depends on the frequency of m and m_j being adjacent in sequences. b^t_{m,m_j} is the transition score between m and m_j that indicates the corresponding feature correlation. σ is leaky ReLU activation function, and tanh () is the hyperbolic tangent activation function. $\mathbf{w} \in \mathcal{R}^{2d''}$, $\mathbf{W} \in \mathcal{R}^{d'' \times d}$, and $\mathbf{c} \in \mathcal{R}^{d''}$ are learnable parameters. $\mathbf{h}^t_m \in \mathcal{R}^{d''}$ and $\mathbf{h}^t_{m_j} \in \mathcal{R}^{d''}$ are the hidden representations of music pieces m and m_j.

Context view. Traditional item-item collaborative filter-based methods perform recommendation according to the similarity between items (music pieces). Generally, each user has specific preferences and items that are interacted with by common users are similar to each other. However, one user may have diverse (global) interest, which may influence the accurate computation of the similarity between music pieces.

Generally, users' contextual interest reflected in session level is relatively more specific than the global interest in their whole historical behavior sequences. Therefore, we try learning the contextual features of music in session level. On HMG, as for music m and its n'' session neighbors $\{s_0, s_1, \ldots, s_{n''-1}\} \subseteq N^c(v_m)$ (sessions containing m), the representation of m in context view is defined as:

$$\mathbf{v}^c_m = \sum_{j=0}^{n''-1} \beta_{m,s_j} \cdot \mathbf{v}'_{s_j}, \tag{26}$$

where $\mathbf{v}'_{s_j}$ is the embedding of m's jth session neighbor s_j, and β_{m,s_j} is the attention weight of s_j for m. Specifically, β_{m,s_j} models the co-occurrence information between the music pieces in session level, and the calculation process of attention weight β_{m,s_j} is similar with Equations (17)–(20), with the music-user edge weight w_{m,u_j} being replaced with music-session weight w_{m,s_j}.

Music view aggregation and update. Then, we need to aggregate the embedding of music pieces learned from HMG in four different aspects, i.e., transition, attribute, context, and interaction views, and update the representation of music in a similar way with user view aggregation and update. Formally, the final embedding of music m is defined via weighted sum strategy as:

$$\mathbf{v}^{agg}_m = \gamma^a \cdot \mathbf{v}^a_m + \gamma^i \cdot \mathbf{v}^i_m + \gamma^t \cdot \mathbf{v}^t_m + \gamma^c \cdot \mathbf{v}^c_m, \tag{27}$$

where $\mathbf{v}^a_m$, $\mathbf{v}^i_m$, $\mathbf{v}^t_m$, and $\mathbf{v}^c_m$ are music m's embeddings in view of attribute, interaction, transition, and context. γ^y is the attention weight of each music view y, which is formally defined as:

$$\gamma^y = softmax\left(r^y\right), \tag{28}$$

$$r^y = \mathbf{q}^\top \cdot \tanh\left(\mathbf{W}^r \cdot \mathbf{v}^y_m + \mathbf{c}^r\right), \tag{29}$$

$$\mathbf{v}^{(l)}_m = (1 - \lambda) \otimes \mathbf{v}^{(l-1)}_m + \lambda \otimes \mathbf{v}^{agg}_m, \tag{30}$$

$$\lambda = \sigma_m\left(\mathbf{W}^\lambda \cdot \left(\mathbf{v}^{(l-1)}_m \middle\| \mathbf{v}^{agg}_m\right) + \mathbf{c}^\lambda\right), \tag{31}$$

where $\mathbf{v}_m^{(l)} \in \mathcal{R}^d$ and $\mathbf{v}_m^{(l-1)} \in \mathcal{R}^d$ are the node representations of user m in the lth and $(l-1)$-th step or layer, respectively. σ_m is a sigmoid activation function, and tanh () is the hyperbolic tangent activation function. $\mathbf{q} \in \mathcal{R}^{d''}$, $\mathbf{W}^r \in \mathcal{R}^{d'' \times d}$, $\mathbf{c} \in \mathcal{R}^{d''}$, $\mathbf{W}^\lambda \in \mathcal{R}^{d \times 2d}$, and $\mathbf{c}^\lambda \in \mathcal{R}^d$ are learnable parameters. Finally, we can obtain the final feature representation (embedding) of each music piece m as $\mathbf{v}_m$.

4.3 Short-term Preference Inferring

Users' short-term preferences play an important role in many sequential recommendation/prediction tasks, including next item recommendation and session-based recommendation. In this work, user's short-term preference is inferred from her/his most recent listening behaviors.

Formally, given user u and her/his historical listening sequence $\{m_0, \ldots, m_{o-i}, \ldots, m_{o-1}\}$, u's short-term preference representation is defined with an attention-based weighted sum strategy as:

$$\mathbf{v}_u^{st} = \sum_{j=o-h}^{o-1} \varepsilon_{u,m_j}^{st} \cdot \mathbf{v}_{m_j}, \tag{32}$$

where $\mathbf{v}_{m_j}$ is the learn embedding of m_j, o is the length of user u's historical sequences, and h is the length of u's recent sequence. Besides ε_{u,m_j}^{st} is the weight of historical record m_j in short-term preference inferring process. Formally, ε_{u,m_j}^{st} is defined with soft-attention mechanism as:

$$\varepsilon_{u,m_j}^{st} = softmax\left(e_{u,m_j}^{st}\right), \tag{33}$$

where e_{u,m_j}^{st} is the score of m_j for u's short-term preference inferring, and it is defined as:

$$e_{u,m_j}^{st} = \mathbf{w}_u^\top \cdot \left(\mathbf{W}_1 \cdot \mathbf{v}_u + \mathbf{W}_2 \cdot \mathbf{v}_{m_j} + \mathbf{W}_3 \cdot \mathbf{v}_{m_{o-1}} + \mathbf{c}\right), \tag{34}$$

where $\mathbf{v}_u \in \mathcal{R}^d$ is the embedding of u, and $\mathbf{v}_{m_{o-1}} \in \mathcal{R}^d$ is the embedding of the most recent music record m_{o-1} in listening sequence. $\mathbf{w}_u \in \mathcal{R}^d$, $\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3 \in \mathcal{R}^{d \times d}$, and $\mathbf{c} \in \mathcal{R}^d$ are learnable parameters.

4.4 Prediction Layer

Generally, users' behaviors depend on their preferences, and we utilize three kinds of preferences of users, i.e., long-term, short-term, and dynamic preferences for the session-based music recommendation task.

Specifically, user u's long-term preference is represented with the learned embedding $\mathbf{v}_u$, and her/his short-term preference is modeled as $\mathbf{v}_u^s$, learned in Section 4.3. Besides, we use the embedding of u's last listening record (music m_o), i.e., $\mathbf{v}_{m_o}$, to model her/his dynamic preference.

Then, we can calculate the interest score of u for candidate music piece $m \in M$ according to user u's preference and music m' feature embedding and then recommend music pieces with high score to u. Formally, the interest score of u for m is defined as:

$$p_{u,m} = \left(\rho^{lt} \cdot \mathbf{v}_u^\top + \rho^{st} \cdot \mathbf{v}_u^{st\top} + \rho^d \cdot \mathbf{v}_{m_o}^\top\right) \cdot \mathbf{v}_m, \tag{35}$$

where $\mathbf{v}_m$ is the embedding of candidate music piece m. Note that several music pieces will be sampled from the music set that have not been listened to by u, and these negative samples are used to help train MEGAN effectively together with the positive example (ground-truth music). ρ^{lt}, ρ^{st}, and ρ^d are weights of long/short-term and dynamic preference, respectively, which are defined as:

$$\left[\rho^{lt}, \rho^{st}, \rho^d\right] = softmax\left(\sigma\left(\mathbf{W}^\rho\left(\mathbf{v}_u \left\|\mathbf{v}_u^{st}\right\| \mathbf{v}_{m_o}\right) + \mathbf{c}^\rho\right)\right), \tag{36}$$

where $\mathbf{W}^\rho \in \mathcal{R}^{3 \times 3d}$ and $\mathbf{c}^\rho \in \mathcal{R}^3$ are learnable parameters.

Table 2. Complete Statistics of Lastfm and 30Music Datasets

Dataset	#(User)	#(Music)	#(Session)	#(Word)	#(Record)	#(Record)/User	#(Record)/Music	Sparsity
Lastfm	900	128,560	73,952	76,982	909,412	1,010	7	99.21%
30Music	2,000	90,868	135,371	55,108	1,492,321	746	16	99.18%

Finally, we can sort the candidate music pieces according to their score in Equation (35) and recommend the music pieces with high score to the target user.

5 EXPERIMENTS

In this section, extensive experiments are conducted on two real-world music listening datasets to answer the following research questions:

RQ1: Does the proposed model MEGAN outperform state-of-the-art baselines in the session-based music recommendation tasks?

RQ2: What are the effects of MEGAN's attention mechanism in the short-term preference inferring and in the prediction layer?

RQ3: How do the parameters, including dimension of embedding and length of historical sequence in MEGAN, affect the recommendation results?

RQ4: Does each view in MEGAN contribute to the performance improvements?

RQ5: How does the proposed model MEGAN perform on datasets with different sparsity and size?

5.1 Experimental Designs

In this section, we introduce the experimental designs in detail, including datasets, baselines, evaluation metrics, and the parameter settings as well as experiment environment.

5.1.1 Dataset. We adopt two real-world datasets, including Lastfm[5] [3] and 30Music[6] [35] to evaluate the proposed approach MEGAN as well as the baselines.

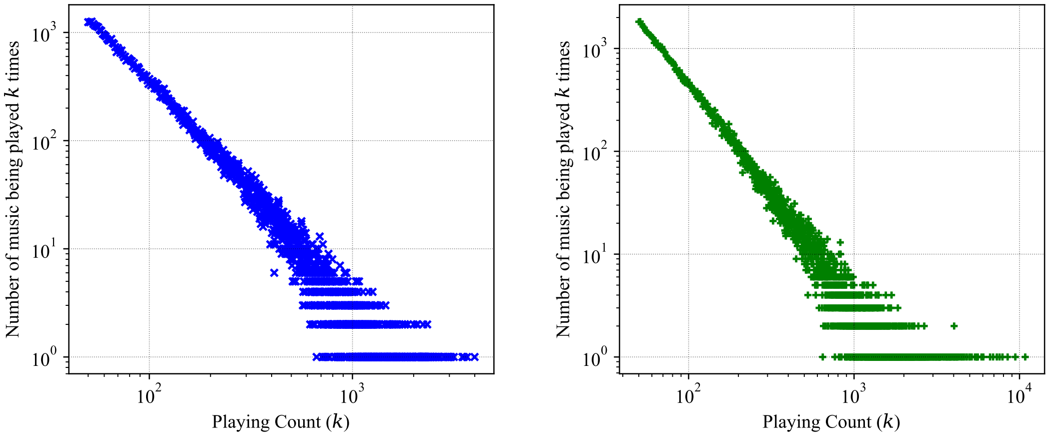
As shown in Table 2, the Lastfm dataset after pre-processing is composed of 909,412 listening records between 900 users and 128,560 pieces of music and the number of the corresponding content words, such as singer, album, tag, and so on, is 76,982. The 30Music dataset includes 1,492,321 interactions to 90,868 music pieces by 2,000 users, and the amount of content words is 55,108. Especially, users' listening sequences are grouped into sessions based on the timestamps. Note that sparsity is the proportion of user-music pair without interaction data. Specifically, the corresponding data sparsity is $1 - (k/(m \times n))$, where k is the amount of all unique records by m users for n music pieces, and $k/(m \times n)$ is the density of dataset.

Figure 4 shows the popularity information of music pieces in two datasets, and the horizontal axis and vertical axis represent the popularity (frequency) k and the amount of music pieces with popularity of k. We can observe the long tail [2] in both datasets. Specifically, only a small number of music pieces are very popular, and the rest are in the heavy tails. Therefore, it is important and meaningful to explore and discover the massive amount of music data via personalized recommender systems.

Besides, the 30Music dataset has more users but less music pieces than the Lastfm dataset, and the two datasets have different statistical characteristics, including the average records per user, the average record per music, and the sparsity. Therefore, these two complementary datasets can be used to evaluate the proposed approach MEGAN and the baselines in a comprehensive way.

[5] http://ocelma.net/MusicRecommendationDataset/lastfm-1K.html.

[6] http://recsys.deib.polimi.it/datasets/.



(a) Lastfm

(b) 30Music

Fig. 4. Popularity analysis of the Lastfm and 30Music datasets.

Table 3. Features Used by the Proposed Approach MEGAN and Baselines

Methods	Feature		
	interaction	listening sequence	content text
Pop	√	×	×
PPop	√	×	×
LineRec	√	×	√
HRM	√	√	×
FPMC	√	√	×
SHAN	√	√	×
RDR	√	√	×
SASRec	√	√	×
HGN	√	√	×
CAME	√	√	√
SRGNN	√	√	×
GCSAN	√	√	×
GCE-GNN	√	√	×
MEGAN	√	√	√

We leave a small percentage of the whole dataset out as validation set, which is used to help to choose the final model. Moreover, the rest dataset is split as training set and testing set, which are two non-overlapping. Specifically, the testing set only includes 20% of users' second half historical music listening sequences, and all the rest data are included in the training set.

5.1.2 Baselines. The proposed approach MEGAN is compared with the following baselines, including basic recommendation methods and state-of-the-art recommendation models. Specifically, the features used by each method are shown in Table 3.

- **Pop** always recommends the music pieces with the highest frequency/popularity in the whole training dataset, and its recommendation results are not personalized.

- **PPop (Personalized Pop)** is similar with Pop, but it recommends the music pieces with the highest frequency in the corresponding user's historical records.
- **LineRec** is based on a network (graph) embedding method [34], and it is used to learn the representation of user and music from Heterogeneous Music Graph for recommendation.
- **HRM** [42] learns the sequential behavior patterns and models users' preferences with the hierarchical representation technique, and it is equipped with two different aggregation strategies, i.e., max pooling (HRM-max) and average pooling (HRM-avg).
- **FPMC** [30] uses **matrix factorization (MF)** and **Markov chains (MC)** to model users' general taste and sequential behavior transition for next-basket recommendation.
- **SHAN** [57] uses a two-layer hierarchical attention to capture the representation of users' long/short-term preferences from their historical behavior sequences for recommendation.
- **RDR** [37] learns the representation of each music piece from users' playing histories via a skip-gram model and obtains users' interest for music recommendation.
- **SASRec** [19] models users' behavior patterns, captures their preferences in a long-term view, and performs recommendation based on the relevant records in users' action history
- **HGN** [26] is a hierarchical gating network with feature, instance, and item gating module that can capture both long- and short-term user preference for recommendation.
- **CAME** [41] learns the dynamic feature representation of music pieces in content and context views and models users' preferences via attention mechanism for session-based music recommendation.
- **SRGNN** [50] is a graph neural networks-based model for complex behavior sequences modeling and session-based recommendation.
- **GC-SAN** [53] is a graph contextualized self-attention model, which utilizes both graph neural network and self-attention mechanism to learn the local dependencies and contextualized representations for session-based recommendation.
- **GCE-GNN** [48] exploits item transitions over all sessions in a more subtle manner for better inferring the user preference of the current session and improving the performance of session-based recommendation.

5.1.3 Evaluation Metrics. In this work, we adopt four metrics, i.e., precision, recall, F1, and **Mean Average Precision (MAP)**, to evaluate the proposed approach MEGAN as well as the baselines comprehensively.

Specifically, precision is the proportion of the relevant music pieces (listened to by the target users) in recommendation list. Formally, precision is defined as

$$P@n = \frac{1}{\#(test)} \sum_{1 \leq i \leq \#(test)} \frac{|R_i \cap G_i|}{|R_i|},$$

where R_i is the ith recommendation list of music pieces by the recommendation model, n is the length of the recommendation list, G_i is the ground-truth list of music pieces that are actually listened to by the target users, and $\#(test)$ is the total amount of all test cases.

Recall is the proportion of the relevant music pieces being successfully recommended by the model, and it is formally defined as

$$R@n = \frac{1}{\#(test)} \sum_{1 \leq i \leq \#(test)} \frac{|R_i \cap G_i|}{|G_i|}.$$

Precision and recall evaluate the recommendation results in different aspects. Specifically, higher precision means that the recommendation models return more relevant music pieces than irrelevant ones, and high recall indicates that the model successfully recommended most of the relevant music pieces.

F1 score is another measure for accuracy evaluation, and it takes both precision and recall into consideration. Formally, F1 score is defined as

$$F1@n = 2 \times \frac{P@n \times R@n}{P@n + R@n}.$$

Moreover, MAP is a widely used metric in information retrieval tasks, and it takes both ranking and accuracy into consideration. Formally, MAP is defined as:

$$M@n = \frac{1}{\#(test)} \sum_{1 \leq i \leq \#(test)} AP_i@n,$$

where $AP_i@n$ is the average precision of the n-item recommendation list in the ith test case, and it is defined as:

$$AP_i@n = \begin{cases} \frac{\sum_{1 \leq j \leq \#(hits_i)} \frac{j}{rank(hits_i(j))} AP_i@n}{|G_i|}, & \#(hits_i) \neq 0 \\ 0, & \#(hits_i) = 0 \end{cases},$$

where $\#(hits_i) = |R_i \cap G_i|$, $rank\ (hits_i\ (j))$ is the rank of the jth relevant music pieces (listened to by the target users) in recommendation list.

5.1.4 Parameter Settings and Experiment Environment. In the proposed model MEGAN, the negative sample amount is set as 5, the batch size is set as 512, and the number of epochs to 10. Besides, the dimension of embedding and the length of historical sequences in short-term preference inferring are set to 256 and 5, respectively, which will be evaluated further in Section 5.3. The Adam optimizer [20] is used to optimize the parameters in MEGAN, where the learning rate is initialized as $3e{-}3$. All the experiments were conducted on a Linux server with Intel(R) Xeon(R) Silver 4108 CPU@1.80 GHz, and a GeForce RTX 2080Ti GPU is used to accelerate the training process. Besides, the versions of Python and PyTorch are 3.6 and 1.5.0, respectively.

5.2 The Comparison with Baselines (RQ1)

In this section, we compare the proposed approach MEGAN with 14 baselines, including traditional recommendation models and deep learning-based recommendation methods, and the results on Lastfm and 30Music datasets are shown in Tables 4 and 5, respectively, where the best results are marked as bold, and the second-best results are underlined. Specifically, we have the following four observations:

First, MEGAN has the best performance in most test cases on Lastfm and 30Music dataset, which shows the effectiveness of MEGAN. Note that MEGAN still achieves the second-best performance in metric of precision@5, although its performance is lower than some baseline in a few cases. Besides, the best results except for MEGAN in different metric/datasets are achieved by several different baselines, while the performance of MEGAN is relatively stable. The improvements show that MEGAN can effectively model and fuse users' dynamic long- and short-term preference for music and perform accurate recommendation, and the multi-view of users and music indeed enhanced MEGAN's ability of learning the accurate representation of users' interest and music pieces' features and improved the performance of session-based music recommendation.

Second, the recommendation methods with attention mechanism or auxiliary information, such as LineRec, SRGNN, CAME, CS-SAN, GCE-GNN, and SASRec, outperform other baselines, and the reason is two-fold. First, attention mechanism can help the recommendation model to focus on the information or features that are important for the recommendation tasks in an adaptive way. Second, the auxiliary information, such as genre and tags of music, can indeed help to learn the

Table 4. Comparisons between MEGAN and Baselines on Lastfm Dataset (mean±s.d.)

Method	P@5	R@5	F1@5	M@5	P@10	R@10	F1@10	M@10
Pop	0.13(±0.00)	0.04(±0.00)	0.06(±0.00)	0.01(±0.00)	0.22(±0.00)	0.13(±0.00)	0.17(±0.00)	0.25(±0.00)
PPop	8.7(±0.00)	2.67(±0.00)	4.08(±0.00)	2.29(±0.00)	7.90(±0.00)	4.84(±0.00)	6.01(±0.00)	3.22(±0.00)
LineRec	13.48(±0.20)	4.13(±0.06)	6.32(±0.09)	5.01(±0.04)	13.57(±0.31)	8.31(±0.22)	10.31(±0.27)	7.41(±0.09)
HRM-max	3.98(±0.05)	1.29(±0.02)	1.95(±0.02)	1.53(±0.03)	3.02(±0.01)	1.95(±0.01)	2.37(±0.01)	1.74(±0.03)
HRM-avg	6.51(±0.20)	2.11(±0.07)	3.18(±0.10)	2.33(±0.08)	5.72(±0.13)	3.70(±0.08)	4.49(±0.10)	2.98(±0.10)
FPMC	3.94(±0.08)	1.21(±0.02)	1.85(±0.04)	1.00(±0.02)	3.69(±0.10)	2.26(±0.06)	2.81(±0.08)	1.40(±0.02)
SHAN	9.09(±0.16)	2.87(±0.04)	4.37(±0.07)	3.28(±0.05)	7.99(±0.12)	5.05(±0.06)	6.19(±0.08)	5.33(±0.05)
RDR	11.67(±0.07)	3.58(±0.02)	5.48(±0.03)	4.54(±0.03)	11.08(±0.03)	6.79(±0.02)	8.42(±0.02)	6.36(±0.03)
SASRec	14.94(±0.08)	4.00(±0.02)	6.31(±0.03)	3.28(±0.02)	13.44(±0.06)	7.19(±0.03)	9.37(±0.04)	5.05(±0.03)
HGN	15.91(±0.19)	1.77(±0.02)	3.18(±0.04)	3.10(±0.04)	14.53(±0.11)	3.23(±0.02)	5.28(±0.04)	4.43(±0.04)
SRGNN	**19.42**(±0.11)	2.16(±0.01)	3.83(±0.02)	5.22(±0.02)	14.32(±0.08)	3.18(±0.03)	5.20(±0.03)	6.27(±0.03)
CAME	10.93(±0.09)	3.37(±0.02)	5.14(±0.03)	4.79(±0.03)	14.36(±0.10)	8.72(±0.05)	10.86(±0.07)	7.57(±0.05)
GC-SAN	14.87(±0.06)	4.56(±0.02)	6.97(±0.03)	**6.16**(±0.05)	11.06(±0.02)	6.78(±0.01)	8.41(±0.01)	7.55(±0.04)
GCE-GNN	15.29(±0.19)	4.69(±0.06)	7.17(±0.09)	5.53(±0.04)	15.36(±0.18)	9.41(±0.11)	11.67(±0.14)	8.60(±0.08)
MEGAN	17.49(±0.08)	**5.36**(±0.02)	**8.21**(±0.04)	5.92(±0.03)	**16.37**(±0.05)	**10.03**(±0.03)	**12.44**(±0.04)	**9.22**(±0.04)
Imprv	−9.10%*	14.29%*	14.50%*	−3.90%*	6.58%*	6.59%*	6.60%*	7.21%*

The best results are marked as bold, and the second-best results are underlined. "Imprv" standards for the improvement achieved by MEGAN than the baseline with the best performance, and "*" indicate the performance differences are statistically significant based on t-test results.

Table 5. Comparisons between MEGAN and Baselines on 30Music Dataset (mean±s.d.)

Method	P@5	R@5	F1@5	M@5	P@10	R@10	F1@10	M@10
Pop	0.25(±0.00)	0.11(±0.00)	0.16(±0.00)	0.06(±0.00)	0.22(±0.00)	0.21(±0.00)	0.21(±0.00)	0.08(±0.00)
PPop	5.57(±0.00)	2.58(±0.00)	3.53(±0.00)	2.92(±0.00)	5.26(±0.00)	4.88(±0.00)	5.06(±0.00)	3.83(±0.00)
LineRec	15.50(±0.15)	7.19(±0.07)	9.82(±0.09)	9.65(±0.11)	15.90(±0.11)	14.75(±0.10)	15.30(±0.11)	13.45(±0.15)
HRM-max	6.81(±0.02)	3.18(±0.01)	4.34(±0.01)	4.16(±0.02)	5.58(±0.02)	5.22(±0.02)	5.40(±0.02)	4.80(±0.02)
HRM-avg	8.66(±0.07)	4.05(±0.03)	5.52(±0.04)	5.14(±0.05)	7.63(±0.09)	7.14(±0.09)	7.37(±0.09)	6.34(±0.08)
FPMC	3.52(±0.07)	1.63(±0.03)	2.23(±0.05)	1.75(±0.02)	3.23(±0.06)	3.00(±0.06)	3.11(±0.06)	2.26(±0.02)
SHAN	11.08(±0.06)	5.14(±0.04)	7.02(±0.05)	6.14(±0.06)	9.74(±0.04)	9.03(±0.06)	9.37(±0.05)	7.84(±0.07)
RDR	13.66(±0.05)	6.34(±0.02)	8.66(±0.03)	8.92(±0.02)	12.82(±0.04)	11.89(±0.04)	12.34(±0.04)	11.56(±0.03)
SASRec	17.40(±0.06)	6.02(±0.02)	8.94(±0.03)	4.76(±0.01)	15.28(±0.05)	10.57(±0.04)	12.50(±0.04)	7.24(±0.02)
HGN	20.27(±0.09)	6.35(±0.03)	9.67(±0.04)	11.60(±0.08)	18.17(±0.09)	11.38(±0.06)	13.99(±0.07)	14.74(±0.12)
SRGNN	**23.74**(±0.05)	7.44(±0.02)	11.33(±0.03)	**16.22**(±0.03)	17.25(±0.09)	10.81(±0.05)	13.29(±0.07)	**17.53**(±0.06)
CAME	12.88(±0.14)	5.90(±0.07)	8.09(±0.07)	9.22(±0.11)	16.24(±0.12)	15.07(±0.11)	15.63(±0.13)	13.36(±0.15)
GC-SAN	18.62(±0.15)	8.64(±0.07)	11.80(±0.09)	11.90(±0.15)	13.94(±0.08)	12.93(±0.08)	13.41(±0.08)	14.27(±0.16)
GCE-GNN	18.44(±0.49)	8.55(±0.22)	11.68(±0.31)	10.96(±0.21)	18.40(±0.27)	17.07(±0.25)	17.71(±0.26)	15.74(±0.24)
MEGAN	21.37(±0.05)	**9.91**(±0.02)	**13.54**(±0.03)	12.27(±0.02)	**19.19**(±0.02)	**17.80**(±0.02)	**18.47**(±0.02)	17.22(±0.02)
Imprv	−9.98%*	14.70%*	14.75%*	−24.35%*	12.75%*	4.29%*	4.29%*	−1.77%*

The best results are marked as bold, and the second-best results are underlined. "Imprv" standards for the improvement achieved by MEGAN than the baseline with the best performance, and "*" indicate the performance differences are statistically significant based on t-test results.

feature representation of music as well as model users' specific preferences, both of which are the key components in the recommender systems.

Third, the deep learning-based models have achieved better performance than other baselines on both datasets. The results show the effectiveness of deep neural networks in capturing high-level features and complex correlations in users' music listening sequences for improving the performance of session-based music recommendation.

Moreover, the Lastfm and 30Music dataset have different statistic characteristics, including the amount of user or music, the average number of records per user or music, the sparsity, and popularity distribution, which also influence the performance of the proposed model MEGAN and baselines. For example, the 30Music dataset has less music pieces (90,868) than the Lastfm dataset (128,560), and the sparsity of 30Music dataset (99.18%) is lower than Lastfm (99.21%). Especially, the average number of records per music on 30Music is larger than it on Lastfm dataset. These statistics show that the recommendation tasks on Lastfm are more challenging than it on 30Music to some extent, so the performance in all four metrics (particularly recall) on 30Music is better than it on Lastfm dataset.

In conclusion, the comparison results with the baselines show the effectiveness of MEGAN in session-based music recommendation tasks.

5.3 The Impacts of Parameters (RQ2)

In this section, we evaluate the influence of two important parameters on the recommendation performance, including the dimension of users' and music pieces' embedding and the length of historical behavior sequences in the inferring process of users' short-term preference in MEGAN.

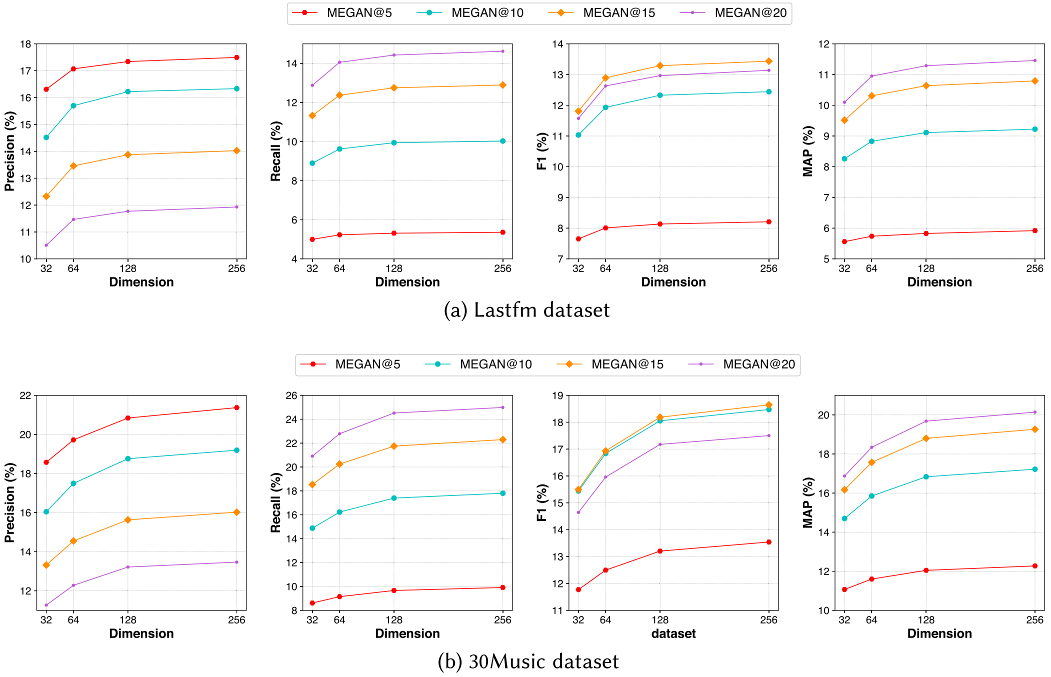
5.3.1 Dimension. The dimension of embeddings is an important factor of the recommendation model's capacity of capturing features and information important for recommendation tasks, and it also influences the time and space complexity of the recommendation model. Specifically, we evaluate the performance of MEGAN with four different dimension settings, i.e., 32, 64, 128, and 256.

As shown in Figure 5, we can observe that MEGAN's performance in all four metrics, i.e., precision, recall, F1, and MAP(@5,10,15,20), is improved when the dimension is increased from 32 to 256 gradually. The results show that the embeddings with higher dimension can indeed capture features and information that are useful for users' preference modeling and music pieces' feature learning and provide informative representation of users and music pieces for recommendation tasks.

Besides, the improvement in all four metrics on 30Music datasets is different with it on Lastfm dataset when we increase the dimension from 32 to 256. We argue that one reason is that the 30Music dataset has different statistic features with the Lastfm dataset. As shown in Table 2, 30Music contains much more users (2,000), music listening records (1,492,321), and sessions (135,371) than Lastfm dataset (900, 909,412, and 73,952, respectively), while the number of music pieces (128,560) and words (76,982) on the latter dataset is larger than the former dataset (90,868 and 55,108, separately).

The performance of MEGAN in terms of all four metrics gets relatively stable when the dimension reaches 256, which means the 256-dimensional embeddings have enough representation capacity for MEGAN on session-based music recomendation tasks. Moreover, further increase of embedding's dimension will also increase the complexity of the recommendation model, such as the amount of learnable parameters. Therefore, we set the dimension of embedding as 256 in consideration of both accuracy and efficiency.

5.3.2 Length of Historical Sequence. In MEGAN, users' short-term preferences are inferred from their recent several music listening behaviors, and the number of records h is an important factor in



(a) Lastfm dataset

(b) 30Music dataset

Fig. 5. Experimental results of the dimension's impact.

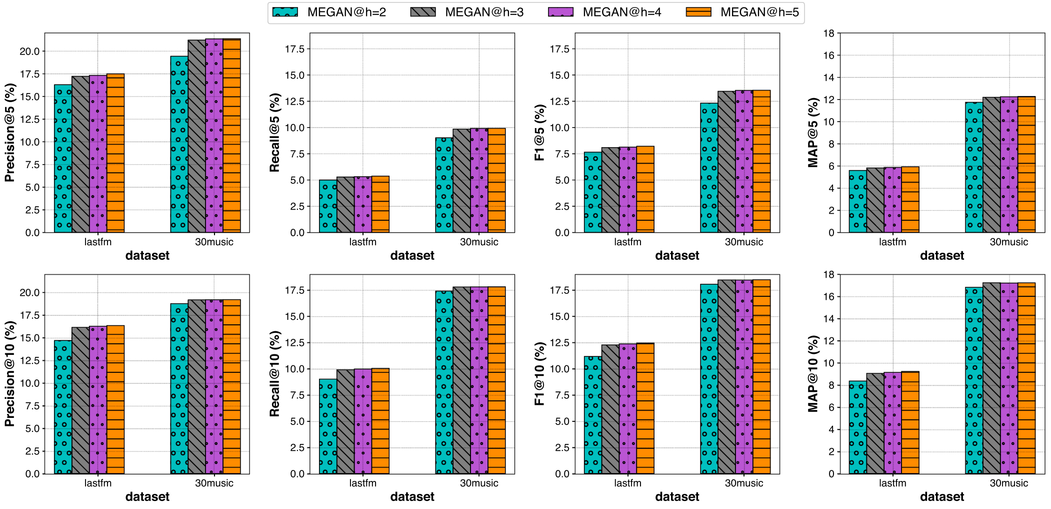


Fig. 6. The impacts of historical sequence length h.

MEGAN. Specifically, larger h can help to model preferences more comprehensively with the risk of noisy data that may adversely influence the performance. In this section, we evaluate the impacts of h's different value settings (2,3,4, and 5) on the performance of music recommendation in detail.

The results are given in Figure 6. We can observe that the performance of MEGAN in metrics of precision, recall, and F1 on two datasets increase as h gets larger. Besides, the performance becomes gradually stable with $h = 4$ or $h = 5$. The results show that longer historical sequences (larger h)

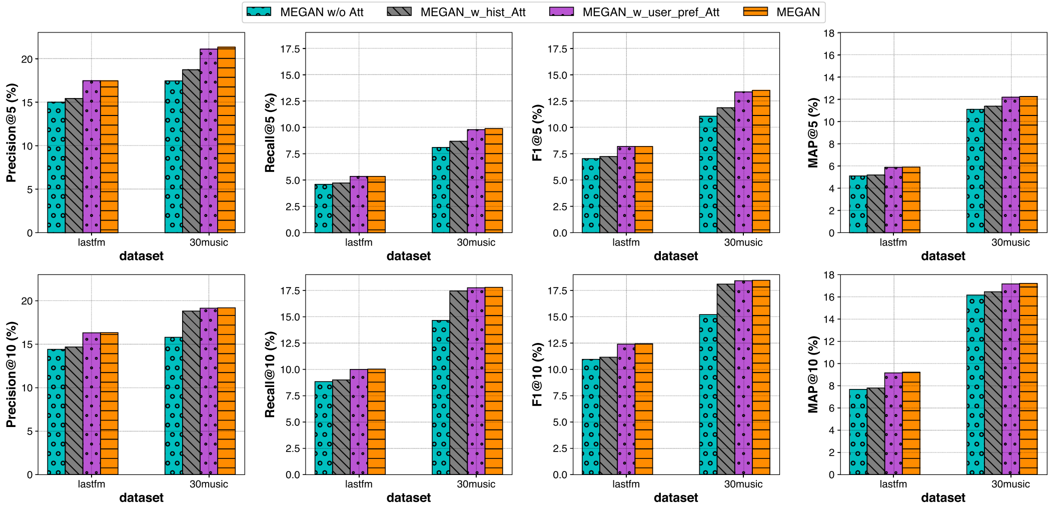


Fig. 7. The impacts of attention mechanism. "MEGAN_w_hist_Att" is the variant of MEGAN' with only the attention in the short-term preference inferring, "MEGAN_w_user_pref_Att" is the variant of MEGAN with only the attention in the prediction layer for user preference fusion (shortly user_pref_attention), and "MEGAN _w/o_Att" is the variant of MEGAN without either hist_attention or user_pref_attention.

provide more behavior data useful for short-term preferences inferring in MEGAN. Especially, the attention mechanism in MEGAN enables it to extract users' key preference that is closely related with the prediction of subsequent listening behaviors while alleviating the influence of noise data in long historical behavior sequences.

Besides, the performance gap between different h is not large, especially when $h \geq 4$, and the reason is two-fold. First, the users' music listening behaviors have fairly high sequentiality, and the most recent behaviors play an important role in sequential/session-based music recommendation tasks. Second, MEGAN combines three different kinds of preferences, i.e., dynamic, long- and short-term, which are complementary with each other for accurate interest modeling and music recommendation.

5.4 The Effects of Attention Mechanism (RQ3)

We adopt attention mechanism in MEGAN for effective behavior patterns modeling and accurate preference representation learning, which is evaluated with ablation experiments of MEGAN and its three variants in this section. Specifically, "MEGAN_w_hist_Att" is the variant of MEGAN' that has only the attention in the short-term preference inferring from users' historical behavior sequences (hist_attention for short), and "MEGAN_w_user_pref_Att" is the variant of MEGAN that only has the attention in the prediction layer for user preference fusion (shortly user_pref_attention). Besides, "MEGAN _w/o_Att" is the variant of MEGAN that does not have either hist_attention or user_pref_attention.

The results are shown in Figure 7, and we can observe that MEGAN outperforms its three variants, and "MEGAN w/o Att" achieves the lowest performance in metrics of precision, recall, and MRR on both Lastfm and 30Music datasets. The results validate the effectiveness of attention mechanism of MEGAN in capturing users' preferences as well as improving the performance of recommendation.

Besides, the performance of "MEGAN_w_user_pref_Att" in all four metrics is quite close to MEGAN, while it is higher than "MEGAN_w_hist_Att." The results show that the

Table 6. The Evaluation of Views in MEGAN

Lastfm dataset								
Method	P@5	R@5	F1@5	M@5	P@10	R@10	F1@10	M@10
MEGAN-w/o-p & a	**17.61**	**5.40**	**8.26**	5.82	**16.45**	**10.08**	**12.50**	9.03
MEGAN-w/o-s & c	16.94	5.19	7.95	5.67	15.48	9.48	11.76	8.61
MEGAN w/o-t	16.89	5.18	7.92	5.76	15.49	9.49	11.77	8.79
MEGAN	17.49	5.36	8.21	**5.92**	16.37	10.03	12.44	**9.22**
30Music dataset								
Method	P@5	R@5	F1@5	M@5	P@10	R@10	F1@10	M@10
MEGAN-w/o-p & a	21.17	9.82	13.41	12.21	19.13	17.74	18.41	17.16
MEGAN-w/o-s & c	21.02	9.75	13.32	12.17	18.82	17.46	18.12	16.93
MEGAN w/o-t	21.03	9.76	13.33	12.22	19.07	17.69	18.35	17.19
MEGAN	**21.37**	**9.91**	**13.54**	**12.27**	**19.19**	**17.80**	**18.47**	**17.22**

The best results are marked as bold, and the second-best results are underlined. "w/o" is the abbreviation for "without." "p & a" represents the profile/attribute view of user/music, and "MEGAN-w/o-p & a" is the variation of MEGAN without textual content data. Similarly, "MEGAN-w/o-s & c" is the variation of MEGAN without session/context view for user/music, and "MEGAN-w/o-t" is the variation of MEGAN without transition view for music.

user_pref_attention for user preference fusion seems to play a more important role in the recommendation tasks than the hist_attention in short-term preference inferring. The reason is two-fold. First, three kinds of user preferences are correlated with each other. The long-term preference and dynamic preference, especially the latter one, can be regarded as an important supplement for short-term preference inferring. Second, the impacts of users' long/short-term and dynamic preferences in the recommendation tasks may change dynamically. For example, users may tend to exploit or explore when they are listening to music, and their behavior patterns and preferences are different in these two situations. Therefore, it is quite important to fuse users' preferences in an appropriate way for accurate music recommendation.

In conclusion, the attention mechanism designed for sequence modeling and preference learning can indeed improve the performance of MEGAN in session-based music recommendation tasks.

5.5 The Effects of View in MEGAN (RQ4)

In the proposed approach MEGAN, the learning process is enhanced by multi-view in HMG, including users' three views (profile/behavior/session) and music pieces' four views (context/interaction/transition/attribute) in an adaptive and effective way. Therefore, we conducted ablation experiments with MEGAN and its three variants, i.e., "MEGAN-w/o-p&a," "MEGAN-w/o-s&c," and "MEGAN-w/o-t," to evaluate the effects of each view. Specifically, "w/o" is the abbreviation for "without." "p&a" represents the profile/attribute views of user/music, and "MEGAN-w/o-p&a" is the variation of MEGAN without textual content data for both user and music. Similarly, "s&c" are the session view of user and the context view of music, and "MEGAN-w/o-s&c" is the variation of MEGAN without session/context view for user/music. Moreover, "MEGAN-w/o-t" is the variation of MEGAN without transition view for music.

The results are shown in Table 6, and we can have the following three observations: (1) MEGAN achieves overall better performance than its variants in consideration of four metrics on Lastfm and 30Music datasets, which verify the effectiveness of the multi-view design in MEGAN. (2) However, MEGAN only achieves the second-best results in some evaluation cases, for example, F1 on Lastfm dataset, where the performance of "MEGAN-w/o-p&a" is slightly better. We argue that the reason is two-fold. First, the content data in profile/attribute views of user/music may contain some noise data and latent semantic features, which should be exploited carefully with advanced strategies.

Table 7. Results of MEGAN on Datasets with Different Sparsity and Size

Dataset	Dataset ID	#(User)	#(Music)	#(Record)	Sparsity	P@5	R@5	F1@5	M@5	P@10	R@10	F1@10	M@10
Lastfm	Lf-v1	900	128,560	909,412	99.21%	17.49	5.36	8.21	5.92	16.37	10.03	12.44	9.22
	Lf-v2	900	423,169	977,332	99.74%	15.81	4.30	6.76	4.94	15.14	8.23	10.66	7.74
30Music	3m-v1	2,000	90,868	1,492,321	99.18%	21.37	9.91	13.54	12.27	19.19	17.80	18.47	17.22
	3m-v2	4,000	156,447	2,736,060	99.56%	21.34	9.62	13.26	11.59	19.46	17.54	18.45	16.65

Second, each view of user and music may play a different role in specific recommendation scenarios. Especially, some tasks rely more on the behavior data, while others may be more dependent on auxiliary information. We will further explore how to incorporate and utilize the heterogeneous information in a more effective and adaptive way in the future works. (3) The performance gap between MEGAN and its three variants is not large, since the other views have complementary effects when some views are missing.

5.6 The Influence of Data Sparsity and Size (RQ5)

We further evaluate the performance of the proposed approach MEGAN on datasets with different sparsity and size, which is quite common in real-world applications. Lf-v1 and 3m-v1 are the datasets used in the previous subsections, and Lf-v2 and 3m-v2 are sparser/larger versions of Lf-v1 and 3m-v1, respectively. The specific results and the statistical information of datasets are listed in Table 7. We can see that MEGAN achieves good performance on Lf-v2 and 3m-v2 datasets, which shows that MEGAN can be effectively applied on larger datasets. Besides, the performance of MEGAN on Lf-v2 and 3m-v2 datasets is not as good as it is on Lf-v1 and 3m-v2, and one reason is that it is more challenging to perform prediction/recommendation on datasets with larger amount of candidate music pieces and higher sparsity.

6 CONCLUSION

In this article, we present a **M**ulti-view **E**nhanced **G**raph **A**ttention **N**etwork, namely, **MEGAN**, which can learn the informative representation of users and music pieces from heterogeneous information based on attention mechanism and graph neural network for session-based music recommendation. Specifically, MEGAN consists of four main steps: behavior and content data modeling with **Heterogeneous Music Graph (HMG), Graph Neural Network (GNN)**-based representation learning on HMG, short-term preference inferring with attention mechanism, and prediction layer that fuses users' preference for music recommendation. Compared with existing methods, MEGAN is capable of: (1) effectively utilizing heterogeneous information including behavior records and content data, (2) learning the preference embedding of users and the feature embedding of music pieces accurately by aggregating neighbors and information from different views in HMG, (3) fusing users' long/short-term and dynamic preferences in an adaptive way for accurate recommendation. Especially, multi-view information is integrated into the representation learning process to enhance MEGAN's ability of capturing the intrinsic features and patterns for accurate music recommendation. Comprehensive experiments are conducted on two real-world music listening datasets, i.e., Lastfm and 30Music, and the comparisons between the proposed model MEGAN and the baselines, including state-of-the-art recommendation models, show that MEGAN achieves better performance in session-based music recommendation tasks.

In the future, we plan to utilize the abundant auxiliary information with knowledge graph [43, 62, 63] and sentiment affective computing techniques for helping to mine the latent semantic or sentiment features of music and inferring users' intent or emotional motivation behind their complex sequential listening data. Besides, we will study on how to further improve the performance of the proposed approach, especially the ranking and the count of relevant music pieces in the recommendation list, via content-driven model [7]. We will also try capturing users' evolving and complex preferences via advanced techniques, for example, dynamic graph neural networks [61] and Hypergraph [21], to further improve the performance of session-based music recommendation models.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* 17, 6 (2005), 734–749. DOI: 10.1109/TKDE.2005.99

[2] Oscar Celma. 2010. The long tail in recommender systems. In *Music Recommendation and Discovery*. Springer, Berlin, 87–107. DOI: 10.1007/978-3-642-13287-2_4

[3] Oscar Celma. 2010. Music recommendation. In *Music Recommendation and Discovery*. Springer, Berlin, 43–85. DOI: 10.1007/978-3-642-13287-2_3

[4] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. 2017. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'17)*. ACM, Association for Computing Machinery, New York, NY, 335–344. DOI: 10.1145/3077136.3080797

[5] Shuhan Chen, Ben Wang, Xiuli Tan, and Xuelong Hu. 2020. Embedding attention and residual network for accurate salient object detection. *IEEE Trans. Cybern.* 50, 5 (2020), 2050–2062. DOI: 10.1109/TCYB.2018.2879859

[6] Tianwen Chen and Raymond Chi-Wing Wong. 2021. An efficient and effective framework for session-based social recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 400–408.

[7] Yashar Deldjoo, Markus Schedl, and Peter Knees. 2021. Content-driven music recommendation: Evolution, state of the art, and challenges. *arXiv preprint arXiv:2107.11803* (2021).

[8] Shuiguang Deng, Dongjing Wang, Xitong Li, and Guandong Xu. 2015. Exploring user emotion in microblogs for music recommendation. *Expert Syst. Applic.* 42, 23 (2015), 9284–9293. DOI: 10.1016/j.eswa.2015.08.029

[9] Mukund Deshpande and George Karypis. 2004. Item-based top-n recommendation algorithms. *ACM Trans. Inf. Syst.* 22, 1 (2004), 143–177. DOI: 10.1145/963770.963776

[10] Jingtao Ding, Guanghui Yu, Yong Li, Xiangnan He, and Depeng Jin. 2020. Improving implicit recommender systems with auxiliary data. *ACM Trans. Inf. Syst.* 38, 1 (2020), 1–27. DOI: 10.1145/3372338

[11] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'16)*. Association for Computing Machinery, New York, NY, 855–864. DOI: 10.1145/2939672.2939754

[12] Lei Guo, Hongzhi Yin, Tong Chen, Xiangliang Zhang, and Kai Zheng. 2021. Hierarchical hyperedge embedding-based representation learning for group recommendation. *ACM Trans. Inf. Syst.* 40, 1 (2021), 1–27. DOI: 10.1145/3457949

[13] Lei Guo, Hongzhi Yin, Qinyong Wang, Tong Chen, Alexander Zhou, and Nguyen Quoc Viet Hung. 2019. Streaming session-based recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1569–1577.

[14] Jiayu Han, Lei Zheng, Yuanbo Xu, Bangzuo Zhang, Fuzhen Zhuang, S. Yu Philip, and Wanli Zuo. 2020. Adaptive deep modeling of users and items using side information for recommendation. *IEEE Trans. Neural Netw. Learn. Syst.* 31, 3 (2020), 737–748. DOI: 10.1109/TNNLS.2019.2909432

[15] Luo He, Hongyan Liu, Yinghui Yang, and Bei Wang. 2021. A multi-attention collaborative deep learning approach for blood pressure prediction. *ACM Trans. Manag. Inf. Syst.* 13, 2 (2021), 1–20. DOI: 10.1145/3471571

[16] Liwei Huang, Yutao Ma, Shibo Wang, and Yanbo Liu. 2021. An attention-based spatiotemporal LSTM network for next POI recommendation. *IEEE Trans. Serv. Comput.* 14, 6 (2021), 1585–1597. DOI: 10.1109/TSC.2019.2918310

[17] Zhenhua Huang, Xin Xu, Honghao Zhu, and MengChu Zhou. 2020. An efficient group recommendation model with multiattention-based neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* 31, 11 (2020), 4461–4474. DOI: 10.1109/TNNLS.2019.2955567

[18] Marius Kaminskas, Francesco Ricci, and Markus Schedl. 2013. Location-aware music recommendation using autotagging and hybrid matching. In *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys'13)*. ACM, Association for Computing Machinery, New York, NY, 17–24. DOI: 10.1145/2507157.2507180

[19] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*. IEEE, 197–206. DOI: 10.1109/ICDM.2018.00035

[20] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR'15)*.

[21] Valerio La Gatta, Vincenzo Moscato, Mirko Pennone, Marco Postiglione, and Giancarlo Sperlí. 2022. Music recommendation via hypergraph embedding. *IEEE Trans. Neural Netw. Learn. Syst.* (2022), 1–13.

[22] Wei-Po Lee, Chun-Ting Chen, Jhih-Yuan Huang, and Jhen-Yi Liang. 2017. A smartphone-based activity-aware system for music streaming recommendation. *Knowl.-based Syst.* 131 (2017), 70–82. DOI: 10.1016/j.knosys.2017.06.002

[23] Xinyi Li, Yifan Chen, Benjamin Pettit, and Maarten De Rijke. 2019. Personalised reranking of paper recommendations using paper content and user behavior. *ACM Trans. Inf. Syst.* 37, 3 (2019), 1–23. DOI: 10.1145/3312528

[24] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. 2015. Gated graph sequence neural networks. In *Proceedings of the 4th International Conference on Learning Representations (ICLR'16).*

[25] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Comput.* 7, 1 (2003), 76–80. DOI: 10.1109/MIC.2003.1167344

[26] Chen Ma, Peng Kang, and Xue Liu. 2019. Hierarchical gating networks for sequential recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'19).* Association for Computing Machinery, New York, NY, 825–833. DOI: 10.1145/3292500.3330984

[27] Sergio Oramas, Vito Claudio Ostuni, Tommaso Di Noia, Xavier Serra, and Eugenio Di Sciascio. 2017. Sound and music recommendation with knowledge graphs. *ACM Trans. Intell. Syst. Technol.* 8, 2 (2017), 21. DOI: 10.1145/2926718

[28] Wenjie Pei, Jie Yang, Zhu Sun, Jie Zhang, Alessandro Bozzon, and David M. J. Tax. 2017. Interacting attention-gated recurrent networks for recommendation. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'17).* Association for Computing Machinery, New York, NY, 1459–1468. DOI: 10.1145/3132847.3133005

[29] Ruihong Qiu, Hongzhi Yin, Zi Huang, and Tong Chen. 2020. GAG: Global attributed graph neural network for streaming session-based recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* 669–678.

[30] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW'10).* ACM, Association for Computing Machinery, New York, NY, 811–820. DOI: 10.1145/1772690.1772773

[31] Diego Sánchez-Moreno, Ana B. Gil González, M. Dolores Muñoz Vicente, Vivian F. López Batista, and María N. Moreno García. 2016. A collaborative filtering method for music recommendation using playing coefficients for artists and users. *Expert Syst. Applic.* 66 (2016), 234–244. DOI: 10.1016/j.eswa.2016.09.019

[32] Jun Song, Jun Xiao, Fei Wu, Haishan Wu, Tong Zhang, Zhongfei Mark Zhang, and Wenwu Zhu. 2017. Hierarchical contextual attention recurrent neural network for map query suggestion. *IEEE Trans. Knowl. Data Eng.* 29, 9 (2017), 1888–1901. DOI: 10.1109/TKDE.2017.2700392

[33] Weiping Song, Zhiping Xiao, Yifan Wang, Laurent Charlin, Ming Zhang, and Jian Tang. 2019. Session-based social recommendation via dynamic graph attention networks. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining (WSDM'19).* Association for Computing Machinery, New York, NY, 555–563. DOI: 10.1145/3289600.3290989

[34] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web (WWW'15).* International World Wide Web Conferences Steering Committee, 1067–1077. DOI: 10.1145/2736277.2741093

[35] Roberto Turrin, Massimo Quadrana, Andrea Condorelli, Roberto Pagano, and Paolo Cremonesi. 2015. 30Music listening and playlists dataset. In *Poster Proceedings of the 9th ACM Conference on Recommender Systems.*

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the Conference on Advances in Neural Information Processing Systems (NIPS'17).* Curran Associates Inc., Red Hook, NY, 6000–6010.

[37] Dongjing Wang, Shuiguang Deng, and Guandong Xu. 2018. Sequence-based context-aware music recommendation. *Inf. Retr. J.* 21, 2-3 (2018), 230–252. DOI: 10.1007/s10791-017-9317-7

[38] Dongjing Wang, Shuiguang Deng, Xin Zhang, and Guandong Xu. 2018. Learning to embed music and metadata for context-aware music recommendation. *World Wide Web* 21, 5 (2018), 1399–1423. DOI: 10.1007/s11280-017-0521-6

[39] Dongjing Wang, Xingliang Wang, Zhengzhe Xiang, Dongjin Yu, Shuiguang Deng, and Guandong Xu. 2021. Attentive sequential model based on graph neural network for next POI recommendation. *World Wide Web* 24, 6 (2021), 2161–2184. DOI: 10.1007/s11280-021-00961-9

[40] Dongjing Wang, Xin Zhang, Zhengzhe Xiang, Dongjin Yu, Guandong Xu, and Shuiguang Deng. 2022. Sequential recommendation based on multivariate Hawkes process embedding with attention. *IEEE Trans. Cybern.* 52, 11 (2022), 11893–11905. DOI: 10.1109/TCYB.2021.3077361

[41] Dongjing Wang, Xin Zhang, Dongjin Yu, Guandong Xu, and Shuiguang Deng. 2020. CAME: Content-and context-aware music embedding for recommendation. *IEEE Trans. Neural Netw. Learn. Syst.* 32, 3 (2020), 1375–1388. DOI: 10.1109/TNNLS.2020.2984665

[42] Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2015. Learning hierarchical representation model for NextBasket recommendation. In *Proceedings of the 38th International ACM SIGIR Conference*

on Research and Development in Information Retrieval (SIGIR'15). ACM, Association for Computing Machinery, New York, NY, 403–412. DOI: 10.1145/2766462.2767694

[43] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'19).* Association for Computing Machinery, New York, NY, 950–958. DOI: 10.1145/3292500.3330989

[44] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19).* Association for Computing Machinery, New York, NY, 165–174. DOI: 10.1145/3331184.3331267

[45] Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021. Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the Web Conference.* 878–887.

[46] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S. Yu. 2019. Heterogeneous graph attention network. In *Proceedings of the World Wide Web Conference (WWW'19).* Association for Computing Machinery, New York, NY, 2022–2032. DOI: 10.1145/3308558.3313562

[47] Xinxi Wang and Ye Wang. 2014. Improving content-based and hybrid music recommendation using deep learning. In *Proceedings of the 22nd ACM International Conference on Multimedia (MM'14).* Association for Computing Machinery, New York, NY, 627–636. DOI: 10.1145/2647868.2654940

[48] Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xian-Ling Mao, and Minghui Qiu. 2020. Global context enhanced graph neural networks for session-based recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* 169–178.

[49] Le Wu, Junwei Li, Peijie Sun, Richang Hong, Yong Ge, and Meng Wang. 2022. DifNet++: A neural influence and interest diffusion network for social recommendation. *IEEE Trans. Knowl. Data Eng.* 34, 10 (2022), 4753–4766.

[50] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence.* 346–353. DOI: 10.1609/aaai.v33i01.3301346

[51] Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang, and Jimeng Sun. 2010. Temporal recommendation on graphs via long-and short-term preference fusion. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'10).* ACM, Association for Computing Machinery, New York, NY, 723–732. DOI: 10.1145/1835804.1835896

[52] Jun Xiao, Hao Ye, Xiangnan He, Hanwang Zhang, Fei Wu, and Tat-Seng Chua. 2017. Attentional factorization machines: Learning the weight of feature interactions via attention networks. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI'17).* AAAI Press, 3119–3125.

[53] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Fuzhen Zhuang, Junhua Fang, and Xiaofang Zhou. 2019. Graph contextualized self-attention network for session-based recommendation. In *Proceedings of the International Joint Conference on Artificial Intelligence.* 3940–3946.

[54] Huiping Yang, Yan Zhao, Jinfu Xia, Bin Yao, Min Zhang, and Kai Zheng. 2019. Music playlist recommendation with long short-term memory. In *Proceedings of the International Conference on Database Systems for Advanced Applications.* Springer International Publishing, Cham, 416–432. DOI: 10.1007/978-3-030-18579-4_25

[55] Jun Yang, Weizhi Ma, Min Zhang, Xin Zhou, Yiqun Liu, and Shaoping Ma. 2021. LegalGNN: Legal information enhanced graph neural network for recommendation. *ACM Trans. Inf. Syst.* 40, 2 (2021), 1–29. DOI: 10.1145/3469887

[56] Tianchi Yang, Linmei Hu, Chuan Shi, Houye Ji, Xiaoli Li, and Liqiang Nie. 2021. HGAT: Heterogeneous graph attention networks for semi-supervised short text classification. *ACM Trans. Inf. Syst.* 39, 3 (2021), 1–29. DOI: 10.1145/3450352

[57] Haochao Ying, Fuzhen Zhuang, Fuzheng Zhang, Yanchi Liu, Guandong Xu, Xing Xie, Hui Xiong, and Jian Wu. 2018. Sequential recommender system based on hierarchical attention network. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'18).* AAAI Press, 3926–3932.

[58] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'18).* Association for Computing Machinery, New York, NY, 974–983. DOI: 10.1145/3219819.3219890

[59] Kazuyoshi Yoshii, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno. 2006. Hybrid collaborative and content-based music recommendation using probabilistic model with latent user preferences. In *Proceedings of the Conference of the International Society for Music Information Retrieval (ISMIR).* 296–301.

[60] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V. Chawla. 2019. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'19).* Association for Computing Machinery, New York, NY, 793–803. DOI: 10.1145/3292500.3330961

[61] Mengqi Zhang, Shu Wu, Xueli Yu, Qiang Liu, and Liang Wang. 2023. Dynamic graph neural networks for sequential recommendation. *IEEE Trans. Knowl. Data Eng.* 35, 5 (2023), 4741–4753. DOI: 10.1109/TKDE.2022.3151618

[62] Rongzhi Zhang, Yulong Gu, Xiaoyu Shen, and Hui Su. 2021. Knowledge-enhanced session-based recommendation with temporal transformer. *arXiv preprint arXiv:2112.08745* (2021).

[63] Yuyue Zhao, Xiang Wang, Jiawei Chen, Yashen Wang, Wei Tang, Xiangnan He, and Haiyong Xie. 2022. Time-aware path reasoning on knowledge graph for recommendation. *ACM Trans. Inf. Syst.* 41, 2 (2022), 1–26. DOI: 10.1145/3531267

[64] Guoqiang Zhong, Haizhen Wang, and Wencong Jiao. 2018. MusicCNNs: A new benchmark on content-based music recommendation. In *Proceedings of the International Conference on Neural Information Processing*. Springer International Publishing, Cham, 394–405. DOI: 10.1007/978-3-030-04167-0_36

Received 7 June 2022; revised 12 February 2023; accepted 12 April 2023