Modeling Sequential Listening Behaviors with Attentive Temporal Point Process for Next and Next New Music Recommendation

Dongjing Wang, *Member, IEEE,* Xin Zhang*, Yao Wan, Dongjin Yu, *Senior Member, IEEE,*
Guandong Xu, *Member, IEEE,* and Shuiguang Deng, *Senior Member, IEEE*

***Abstract*—Recommender systems, which aim to provide personalized suggestions for users, have proven to be an effective approach to cope with the information overload problem existing in many online applications and services. In this paper, we target on two specific sequential recommendation tasks: *next music recommendation and next new music recommendation*, to predict next (new) music piece that users would like based on their historical listening records. In current music recommender systems, various kinds of auxiliary/side information, e.g., item contents and users' contexts, have been taken into account to facilitate the user/item preferences modeling, and have yielded comparable performance improvement. Despite the gained benefits, it is still a challenging and important problem to fully exploit the sequential music listening records, due to the complexity and diversity of interactions and temporal contexts among users and music, as well as the dynamics of users' preferences. To this end, this paper proposes a novel Attentive Temporal Point Process (ATPP) approach for sequential music recommendation, which is mainly composed of a temporal point process model and an attention mechanism. Our ATPP can effectively capture the long- and short-term preferences from the sequential behaviors of users for sequential music recommendation. Specifically, ATPP is able to discover the complex sequential patterns among the interaction between users and music with the temporal point process, as well as model the dynamic impact of historical music listening records on next (new) music pieces adaptively with an attention mechanism. Comprehensive experiments on four real-world music datasets demonstrate that the proposed approach ATPP outperforms state-of-the-art baselines in both next and next new music recommendation tasks.**

***Index Terms*—recommender system, temporal point process, attention mechanism, user profiling, music recommendation**

I. Introduction

The development of web and mobile technologies are leading to a rapid growth of digital contents production (e.g., digital music and micro videos). For example, in the digital media market, users can access to more than 75 million digital songs in Apple iTunes[1], and 75 million songs in Amazon music[2] (statistics in August 2021). However, the flooding musical contents make it difficult for users to find music pieces that meet their preferences, which is called information overload problem. Recommender systems [1], as one of the most successful applications of data mining and machine learning in practice, have proven to be an effective approach to alleviate the information overload problem by providing personalized contents/services from enormous accessible data.

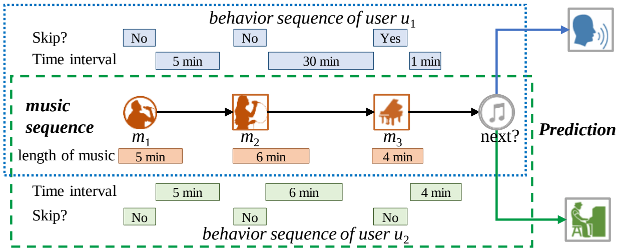


Figure 1. Given the same music pieces with two users' different listening behaviors, the appropriate recommendation results may be different. Here, "*interval*" is the time span between timestamps of two adjacent music records in listening sequences, and "*skip?*" indicates whether the user has skipped to next music piece before the end of current one.

Existing efforts. Currently, recommendation methods, including collaborative filtering (CF) [2], content-based methods [3], context-aware methods [4] and their hybrid ones [5], have been successfully applied in many fields, such as movie/video recommendation [6], [7], point of interests (POI) recommendation [8]. Especially, research in music recommender systems (MRSs) [9] has recently experienced a substantial gain in interest both in academia and in industry. However, in many music websites and applications, the interactions between users and music pieces are recorded over time as music listening/playing sequences, which cannot be fully exploited by traditional CF or content-based methods. Therefore, sequential recommendation methods [10], [11] are proposed to incorporate the sequential information and provide real-time recommendations appropriate for users' current context to promote their experiences.

A motivating scenario. As shown in Figure 1, we use a toy example to describe the recommendation scenario studied in this paper as well as our motivation. Specifically, two users' listening sequences on the same music set may yield different

D. Wang is with School of Computer Science and Technology, Hangzhou Dianzi University, China. e-mail: (Dongjing.Wang@hdu.edu.cn).

X. Zhang*, corresponding author, is with School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, 310018, China. e-mail: (zhangxin@hdu.edu.cn).

Y. Wan is with School of Computer Science and Technology, Huazhong University of Science and Technology, China. e-mail: (wanyao@hust.edu.cn).

D. Yu is with School of Computer Science and Technology, Hangzhou Dianzi University, China. e-mail: (yudj@hdu.edu.cn).

G. Xu is with Advanced Analytics Institute, University of Technology Sydney, Australia. e-mail: (Guandong.Xu@uts.edu.au)

S. Deng is with School of Computer Science and Technology, Zhejiang University, China. e-mail: (dengsg@zju.edu.cn)

[1] https://www.apple.com/my/apple-music/

[2] https://www.amazon.com/music/unlimited

recommendation results, which depends on their preferences inferred from their listening behaviors. For example, in user u_1's sequence (blue), the most recent music piece m_3 generally will have more influence on the prediction of target music piece than m_1 and m_2. However, u_1 skipped m_3 to next music piece before the end of m_3, which indicates that u_1 may not be interested in m_3. Therefore, the prediction of next music piece for u_1 mainly depends on m_1 and m_2, and the result is mostly likely to be vocal music. Note that "*skip*" behaviors can be inferred from users' music listening sequences via comparing the listening time and the length of music. For example, m_3's full length is 4 minutes, but u_1 listened to m_3 for only 1 minute. Therefore, we can infer that u_1 skipped m_3. Besides, as for the second sequence (green), user u_2 has listened to vocal music and instrumental music, so he/she may enjoy music pieces with these two genres next. In brief, it is essential to fully utilize the temporal context and sequential listening behavior in users' historical records for better recommendation.

New challenges. As for the music recommendation scenario mentioned above, existing sequential recommendation methods still face the following three challenges:

(1) *How to capture and leverage users' long- and short-term preferences for better recommendation?* Especially, music pieces are not neutral items but carriers of emotions and thoughts, and users' preferences for music may change frequently. For example, a user may prefer rock music when doing exercise (short-term preferences), though he/she likes pop music better in general (long-term preference).

(2) *How to fully exploit users' behaviors and temporal context in music listening sequences?* Users' music listening sequences contain important information that is useful for recommendation/prediction tasks. For example, short interval between two listening records may indicate strong correlations between them, which is a kind of sequential listening patterns. Besides, the "*skip*" behaviors can help to capture users' preferences accurately.

(3) *How to model the complex transitions and correlations between users and music pieces?* Users' next listening behaviors are associated with their previous listening records. Especially, a user probably listens to some music pieces with same or similar genre together, which can help to model users' preferences effectively and capture music pieces' features accurately.

Therefore, it is essential to fully exploit users' music listening sequences to capture users' real-time preferences accurately for better music recommendation. Besides, different from many scenarios, such as movie or book, where users hardly rate the same item for more than one time, users may listen to the same music pieces repeatedly in music applications. Therefore, the next music piece has a considerable probability to be new for the user in sequential music recommendation, resulting in a more challenging and important task of next new music recommendation. Especially, this task can help users to explore interesting new music pieces.

Our solution and contributions. In this paper, we propose a novel sequential music recommendation method named Attentive Temporal Point Process (ATPP) to predict next (new) music that a user will likely listen to in a "*near future*". As shown in Figure 2, ATPP consists of three modules: sequence modeling, preference capturing, and recommendation. Specifically, ATPP firstly learns users' sequential patterns from their music listening sequences accurately and models the complex correlations between music pieces as well as temporal context information with a temporal point process (TPP) (addressing challenge 2 and 3). Especially, TPP can effectively model the time dependency and feature interactions between music pieces in listening sequences, and help learn users' listening patterns. Then, a self-attention mechanism is designed to enhance ATPP in modeling dynamic impact of historical listening records on next music piece (addressing challenge 3) and leveraging users' long- and short-term preferences (addressing challenge 1). Note that the adopted attention model can increase the adaptivity of ATPP by automatically calculating the weights of different listening records in sequences, and help capture users' preferences accurately. Finally, we can perform next and next new music prediction and recommendation according users' real-time preferences, which are obtained from their historical listening sequences with ATPP.

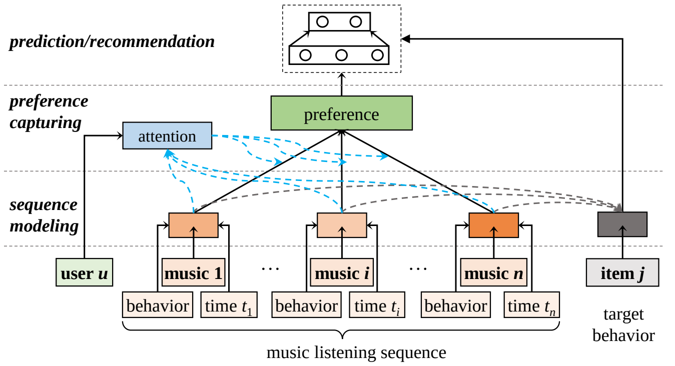


Figure 2. The framework of ATPP in brief

Compared with existing methods, the proposed approach ATPP is capable of: 1) fully exploiting users' listening behaviors and temporal context; 2) precisely modeling dynamic relevance and complex relationships between music pieces in listening sequences; and 3) effectively capturing and leveraging users' long/short-term user preferences for sequential music recommendation. To summarize, the main contributions of this paper are as follows:

- We devise a temporal point process based method to learn the complex relationships between music pieces in users' listening sequences, and then infer and model their long/short-term preferences with an attention model.
- We propose a sequential music recommendation model named ATPP, which can recommend appropriate music pieces based on target users' long/short-term preferences.
- Extensive experiments conducted on four music datasets show that ATPP outperforms state-of-the-art baselines in both next and next new music recommendation tasks.

Organization. The rest of this paper is organized as follows. Section II describes the related works. In Section III and Section IV, we introduce the problem definitions and the proposed model in detail. Then, extensive experiments of the proposed approach are given in Section V. Finally, the conclusion and future works are provided in Section VI.

II. Related Work

In this section, we mainly investigate some related works about music recommendation and sequential recommendation. Besides, we also introduce some related works that inspire this study, such as temporal point process and attention mechanism, as well as their applications.

A. Music Recommendation

Generally, existing works on music recommendation [9], [12] mainly fall into four categories: collaborative filtering (CF) methods, content-based recommendation methods, context-aware methods, and hybrid recommendation methods. Specifically, collaborative filtering based music recommendation approaches [13] can be further categorized into user-based CF (UCF) and item-based CF (ICF). On the other hand, content-based recommendation methods [14], [15] perform recommendation based on users' profiles and music pieces' acoustic feature or textual metadata, such as tags, lyrics, and so on. Cheng et al. [16] explore how acoustic similarity can be used to improve music recommendation, especially for songs from these lesser known artists. Context-aware recommender systems [4], [17] incorporate the contexts related to environments or users to achieve better recommendation performance. Generally, the contexts include temporal information [18], geographical location [19], users' emotional state [20], [21] and so on. Zangerle et al. [22] analyze the connection between users' emotional states reflected in tweets and their musical choices, and propose music ranking strategies that incorporate users' musical preferences, affective information and hashtag contexts. Recently, more and more works have tried to combine different kinds of recommendation methods to alleviate the influence of data sparsity and further improve the performance of recommendation, which belong to hybrid methods [23], [24]. Besides, many existing works have studied users' listening behavior in various aspects, which also promotes the research on music recommendation. For example, Lee et al. [25] investigate the contexts in which music recommendations occur, in order to help understand the impact of music recommendations on people's lives and social relationships (and vice versa). Manolovitz et al. [26] find that the more times a listener is exposed to a song, the more likely she is to return to the song.

B. Sequential Recommendation

Users' interactive actions/events recorded in online web applications and systems play an important role in understanding their underlying requirements and mining behavior patterns, and lots of methods have been developed to model users' sequential behaviors for prediction or recommendation. For example, Rendle et al. [27] propose the factorized personalized Markov chains model, which combines first-order Markov chains with matrix factorization technique for recommendation. Wang et al. [28] propose a hierarchical representation model to model complicated interactions between users and items for the task of next basket recommendation. However, these methods mainly focus on mining the local sequential patterns between adjacent interaction records, ignoring the long-term dependence in users' behavior sequences.

In addition to traditional sequential recommendation methods, the rapid development of deep learning promotes its widely applications in sequence modeling, prediction, recommendation, and so on. For example, Hidasi et al. [29] apply recurrent neural networks (RNN) on session-based recommender systems. Zhu et al. [30] propose a variant of Long Short-Term Memory (LSTM), named Time-LSTM, to model users' actions as time series for recommendation. Ying et al. [11] use a hierarchical attention network to model users' behavior patterns and capture their long/short-term preferences for sequential recommendation. Zhao et al. [31] propose a Spatio-Temporal Gated Network (STGN) to model personalized sequential patterns as well as rich context for users' long/short-term preferences modeling and recommendation. Besides, some existing works focus on understanding users' playlists or listening sequence, which inspire many sequential recommendation models. Vall et al. [32] propose a hybrid recommender system that integrating the collaborative information in music playlists with song feature for automated music playlist continuation. Tang et al. [33] propose a unified and flexible model named Convolutional Sequence Embedding Recommendation (Caser) to learn users' general preferences from their behavior sequences and capture their sequential behavior patterns for sequential recommendation. Yuan et al. [34] present a efficient and effective convolutional generative model NextItNet for session-based top-N item recommendations.Kang et al. [35] address the next item recommendation task with a self-attention based sequential model (SASRec) Ma et al. [36] combine hierarchical gating network with the Bayesian Personalized Ranking (BPR) to capture users' long/short-term interests for the sequential recommendation. Sun et al. [37] propose a Transformer based sequential recommendation model called BERT4Rec, which employs the deep bidirectional self-attention to model user behavior sequences.

C. Temporal Point Process

Temporal point process (TPP) is generally used to model the probability of event/item in sequences and learn their correlations. Compared with RNN/LSTM-based method, TPP-based method can explicitly incorporate the time information, including timestamp and interval, and models sequences as well as the correlations between events in sequences via conditional intensity function in a probabilistic perspective. As one typical variant of TPP, Hawkes process [38] models the occurrence of future event based on past events, and it assumes that past events can temporarily raise the probability of future events, which is known as self-exciting effect. Hawkes process assumes that the excitation is positive, additive over the past events, and exponentially decaying with time. However, in practice, the occurrence of one event may inhibits another one, which violates these assumptions. To mitigate this issue, Rotondi et al. [39] present self-correcting process, which adopts an ever-increasing probability for the target event, and the occurrence of other events will reduce the probability by a certain amount. Du et al. [40] combine TPP with recurrent neural network, and proposed a method named Recurrent Marked Temporal Point Process (RMTPP) to predict the time of next event occurrence and the corresponding marker. Mei et al. [41] propose a new variant of TPP named neural Hawkes process (NHP) based on a self-modulating multivariate point process and a novel continuous-time LSTM. Since event data become more and more pervasive, TPP and its variants have been widely used in many applications, such as online advertisement [42], prediction and detection [43], [44], and so on. For example, Xu et al. [43] propose a framework for modeling the transition events of patient flow via TPP. Dutta et al. [44] combine Hawkes process with topic model, and present a novel fake retweeters detector named HawkesEye, which can exploit textual content data and time information for better detection performance.

D. Attention Mechanism

As one important technique in deep learning, attention mechanism [45]has been applied in many applications, such as computer vision and natural language processing [46], query suggestion [47], prediction and recommendation [48], [49], and so on. Recently, there are many attempts of applying attention mechanism in recommendation. For example, Li et al. [50] explore a hybrid encoder with an attention mechanism to model the users' sequential behaviors and capture their main purpose in the current session for accurate sequential recommendation. Chen et al. [51] incorporate implicit feedback into a Collaborative Filtering (CF) framework together and combine them with an attention model in both item-level and component-level for accurate multimedia recommendation. Xiao et al. [52] propose Attentional Factorization Machines, which combines attention model and Factorization Machine (FM) to measure the significance and relevance between different features as well as their interactions. Wang et al. [49] present a content- and context-aware music recommendation method namely CAME based on network embedding with attention and Convolutional Neural Network, which can cope with various dynamic features of music. Han et al. [53] propose a deep neural networks based recommendation framework to learn the adaptive representations of users. Li et al. [54] explicitly model the absolute positions of items and the time intervals with self-attention mechanism for next item prediction.

III. Problem Formulation

We start by giving a formal description of the studied problem setting with some definitions. The notations and symbols used in this paper are summarized in Table I.

Definition 3.1: ***Music Listening Record.*** Let $U = \{u_1, u_2, \ldots, u_{|U|}\}$ denote the whole user set, $M = \{m_1, m_2, \ldots, m_{|M|}\}$ represent the item set, and T is the time domain. A music listening record r is a tuple $(u, m, t, l_m) \in U \times M \times T \times L$, which represents the interaction record between user u and music piece m at time t, and l_m is the length of music m.

Table I
Symbols used in this work

Symbol	Description
$u \in U$	A user u in the user set U
$m \in M$	A piece of music m in the music set M
H_u	u's music listening sequence
$H_{u,t} \subseteq H_u$	u's historical music listening sequence before time t
$\mathbf{U} \in \mathcal{R}^{\|U\| \times d}$, $\mathbf{V} \in \mathcal{R}^{\|M\| \times d}$	User embedding matrix and music embedding matrix
$\mathbf{v} \in \mathcal{R}^d$	The d-dimension feature vector representation (embedding) of user or music
l_m	The full length of music m
$\omega_{h,m}$	The degree to which historical music piece $h \in M$ initially excites the target music $m \in M$
$\kappa(\cdot)$	The kernel function that incorporates temporal and behavior information in music listening sequences
$\kappa_t(\cdot)$, $\kappa_b(\cdot)$	The temporal influence function and the behavior modeling function
$\delta_u \geq 0$	The decay rate of historical influence
$\mathbf{W}_l$, $\mathbf{W}_s$, $\mathbf{W}_{\bar{l}}$	Model parameters
$\mathbf{A} \in \mathcal{R}^d \times d$	The transition matrix
$f(\cdot)$, $f'(\cdot)$	Mapping functions
$\tilde{\lambda}_{m\|u}(t)$	The predicted preference of u for music m at time t
$>_{u,t}$	The ranking of candidate music for user u at time t

Definition 3.2: ***Music Listening Sequence.*** Let H be the collection of all users' historical music listening records and $u \in U$ is a user. Then, u's historical music listening sequence is defined as

$$H_u := [(u, m_1, t_1, l_{m_1}), (u, m_2, t_2, l_{m_2}), \ldots,$$
$$(u, m_{|H_u|-1}, t_{|H_u|-1}, l_{m_{|H_u|-1}})] .$$

Note that we can infer important behavior information from H_u. For example, time interval can be obtained from timestamps t of records in music listening sequences. Besides, we can also infer whether a user skips to next music piece before the end of currently listening one according to time interval and length of music.

Given a user $u \in U$ associated with her/his historical music listening sequence $H_{u,t} \subseteq H_u$ until time t, our goal is to predict u's preference for music m at $t \in M$ and perform next music recommendation. However, the next music piece has a considerable probability to be new for the given user, which results in a more challenging and important task of next new music recommendation. In this work, both next music recommendation and next new music recommendation tasks are taken into consideration. Note that there is only one correct answer in each of those two tasks.

IV. Methodology

The basic idea of Attentive Temporal Point Process (ATPP) is to learn the correlations among music pieces and behavior patterns from users' music listening sequences for accurate music prediction and recommendation. Figure 3 presents the workflow of ATPP, which is composed of three main components: 1) sequence modeling, 2) preference capturing, and 3) music recommendation.

Specifically, given a sequence of users' music listening records, ATPP firstly models music listening sequence with multivariate temporal point process, and embeds the music

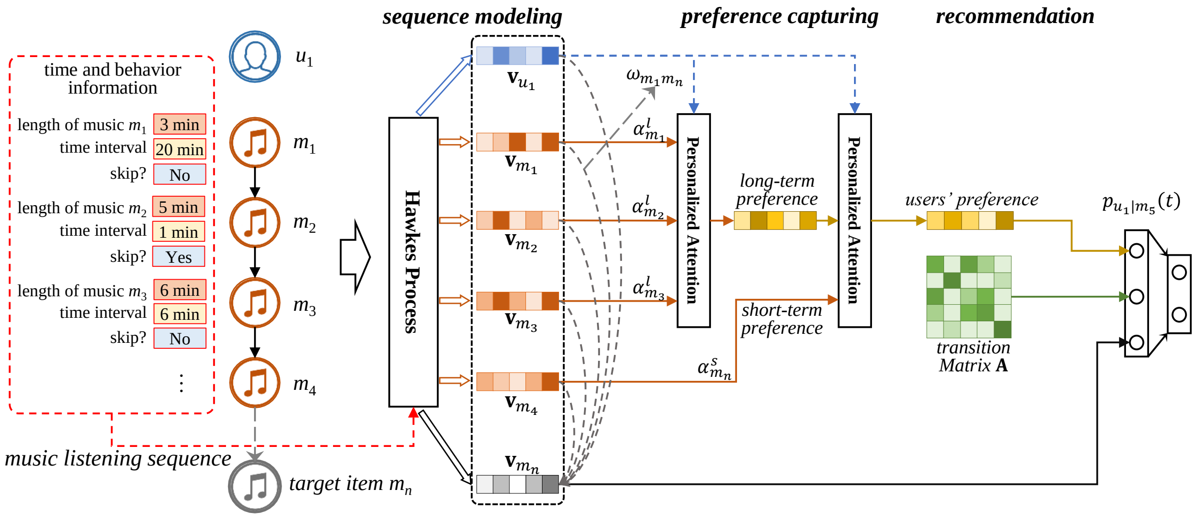


Figure 3. The workflow of the proposed model ATPP. ATPP consists of three steps: 1) modeling users' music listening sequences and time information, 2) exploiting users' complex behavior information and capturing their long/short-term musical preferences, and 3) performing next and next new music recommendation based on users' preferences and transition matrix.

pieces into a low-dimensional space. That is, users and items are represented as low dimensional denser feature vectors (embeddings), which are more informative and effective than the users/items' id or one-hot representations. Then, ATPP adopts a self-attention mechanism to capture the complex correlations between music pieces and learn users' preferences for music by leveraging users' long- and short-term interest. Finally, ATPP employs a transition matrix to model users' preferences and music pieces' feature, and performs next (new) music recommendation based on users' preferences obtained from their historical behavior sequences with ATPP. Next, we will elaborate the details of each component.

A. Music Listening Sequence Modeling

In the proposed model, users' music listening sequences are modeled with a devised Temporal Point Process (TPP). Specifically, the TPP can model records (music listening events) in behavior sequences in continuous time space by learning the time dependency between events. Formally, TPP represents the probability of a record occurs at time t (more precisely, in the infinitesimally wide interval $[t, t+\Delta t)$) as $\lambda(t)\,\Delta t$. Specifically, $\lambda(t) \geq 0$ is known as the intensity function, which represents the arrival rate of sequential listening records.

Generally, the prediction of target music piece depends on users' historical music listening behaviors. In this work, a record (u, m, t, l_m) indicates that a user u has listened to music m at time $t \in \mathcal{R}^+$ (a set of non-negative real numbers). Besides, l_m is m's length, which will be explained in the following part. Formally, given user $u \in U$ as well as u's historical listening sequence H_u, the conditional intensity function for the arrival of target music $m \in M$ at time t is formally defined as follows:

$$\tilde{\lambda}_{m|u}(t) = \sum_{h\in H_{u,t}} \omega_{h,m}\kappa(t-t_h, l_h), \qquad (1)$$

where $H_{u,t} \in H_u$ denotes u's recent historical behavior sequence before time t, t_h is the timestamp of music piece h, and l_h it the full length of h. $\omega_{h,m}$ represents the degree to which historical music piece h initially excites target music piece m. $\kappa(\cdot)$ is a kernel function that incorporates time interval and users' listening behaviors into sequence modeling.

Specifically, recent music pieces in H_u generally will have more influence on the prediction of next music piece than others. Therefore, smaller interval $|t-t_h|$ indicates stronger impact of h on m. Besides, we argue that the correlation between h and m also depends on whether u skips h before the end of it. For example, u might listen to h, whose full length l_h is 6 minutes, for only 1 minute before skipping to next music. In this case, the correlation between h and m is not strong, and h plays a very weak role in u's listening sequence modeling or the prediction of next music. Note that we can infer that user may dislike some genres if he/she skips some music pieces, and it is also important information for preference capturing and recommendation tasks that how long the user have listend to the music pieces that he/she skipped. Therefore, the skip behavior is quite important for improving recommendation performance, and it is incorporated in the proposed approach instead of being filtered in the pre-processing stage. Formally, $\kappa(\cdot)$ consists of two kernel functions, which are defined as:

$$\kappa(t-t_h, l_h) = \kappa_t(t-t_h)\,\kappa_b(l_h), \qquad (2)$$

where $\kappa_t(t-t_h) = \exp(-\delta_u(t-t_h))$ calculates the influence of historical records, which decays exponentially with time. Especially, $\delta_u \geq 0$ denotes the decay rate of historical influence, and it is a user-dependent (personalized) parameter since each user's preference may decay in different rates. Besides, $\kappa_b(l_h)$ incorporates users' behavior information, which is defined as:

$$\kappa_b(l_h) = \begin{cases} \exp\left(-(l_h - \Delta t_h)/l_h\right), & if \Delta t_h < l_h \\ 1, \ if \ \ (t-t_h) \geq l_h \end{cases}, \qquad (3)$$

where Δt_h it the time interval between h and its subsequent music piece in u's listening sequence.

In a word, two terms in Equation (2) model temporal context and behaviors in user u's music listening sequence, respectively.

Specifically, when a user listens to a piece of music, the intensities of all music pieces are elevated or inhibited by certain degree, which depends on the time interval between them and u's specific listening behaviors, i.e. skipping current music or not.

In traditional temporal point process, the inputs are sequences of the original user or item IDs represented as one-hot vecotrs, whose dimension is the same as the size of item set. However, one-hot vectors suffers from serious dimensional disaster and data sparsity problems, especially when the size of item set reach millions or even larger. Besides, one-hot vectors has very limited representation capacity, because it cannot fully capture the intrinsic features of music pieces or their correlations.

In this work, the proposed model ATPP can learn the informative low-dimensional features (embeddings) of users and music pieces, which capture both items' features and relationships in users' listening sequences. Formally, each music piece $m \in M$ in the behavior sequences is transformed into corresponding feature embeddings $\mathbf{v}_m \in \mathcal{R}^d$ with an music embedding matrix $\mathbf{V} \in \mathcal{R}^{|M| \times d}$, where d represents the music embeddings' dimension of and M is the music set. Similarly, user u's preference embedding $\mathbf{v}_u \in \mathcal{R}^d$ can be obtained by looking up a user embedding matrix $\mathbf{U} \in \mathcal{R}^{|U| \times d}$.

Although recommendation methods like matrix factorization or latent factor models [55] can also learn the feature vectors of users and items, ATPP can capture more high-level dynamic key features and sequential patterns via temporal point process. Then, we can feed the d-dimensional embeddings of music pieces into the intensity function in Equation (1). Specifically, $\omega_{h,m}$, the degree to which historical music h in a listening sequence initially excites current music m, depends on features of h and m. Formally, $\omega_{h,m}$ is a mapping function $f(\cdot) : \mathcal{R}^d \times \mathcal{R}^d \to \mathcal{R}$, which can be defined as cosine similarity, dot product, or negative Euclidean distance between vectors.

B. Long/Short-Term Preferences Capturing

In the devised multivariate temporal point process, $\omega_{h,m}$ depends on features of h and m. However, we argue that the correlations between music pieces may be different for each user. For example, some users may have relatively stable preferences for music, and their listening behaviors are more repetitive and predictive. Therefore, their long-term preferences reflected in their listening sequences have large impact on the prediction of target music piece. On the other hand, some users may have relatively diverse preferences, and the music pieces they listen to change frequently. Then, in this case, the prediction of target music mainly depends users' short-term dynamic preferences, which can be inferred from recent listening records. Formally, as for a user u's recent music listening sequence $H_u = \{m_1, m_2, \ldots, m_n\}$ (only retain music id for simplicity and $n = |H_u|$), we use the last clicked music piece m_n to obtain u's short-term preferences, and infer her/his long-term preferences from the rest music listening records $H_u \backslash m_n = \{m_1, m_2, \ldots, m_{n-1}\}$.

Firstly, we calculate the long-term personalized weight α^l of music piece m_i in $H_u \backslash m_n$ given user u. Specifically, α^l is a user and historical item dependent parameter, which can be formally defined with self-attention mechanism [45], [52] as:

$$\alpha^l_{m_i} = \frac{\exp\left(f'\left(\mathbf{v}_u, \mathbf{W}_l \mathbf{v}_{m_i}\right)\right)}{\sum_{h \in H_u \backslash m_n} \exp\left(f'\left(\mathbf{v}_u, \mathbf{W}_l \mathbf{v}_h\right)\right)}, \quad (4)$$

where $\mathbf{v}_u \in \mathcal{R}^d$ and $\mathbf{v}_m \in \mathcal{R}^d$ are embeddings of user u and music m, respectively, d is the dimension of embedding, $f'(\cdot)$ is negative squared Euclidean distance between vectors, and $\mathbf{W}_l$ is a model parameter.

Then, we can define the short-term personalized weight α^s of last clicked music piece m_n as follows:

$$\alpha^s_{m_n} = \frac{\exp\left(f'\left(\mathbf{v}_u, \mathbf{W}_s \mathbf{v}_{m_n}\right)\right)}{\exp\left(f'\left(\mathbf{v}_u, \mathbf{W}_s \mathbf{v}_{m_n}\right)\right) + \exp\left(f'\left(\mathbf{v}_u, \mathbf{W}_{\bar{l}} \mathbf{v}_{\bar{l}}\right)\right)}, \quad (5)$$

where $\mathbf{W}_s$, $\mathbf{W}_{\bar{l}}$ are model parameters, and $\mathbf{v}_{\bar{l}}$ is weighted averaged long-term embedding, which is defined as:

$$\mathbf{v}_{\bar{l}} = \frac{1}{n-1} \sum_{h \in H_u \backslash m_n} \alpha^l_{m_i} \mathbf{v}_{m_i}. \quad (6)$$

Then, we can reformulate the degree of impacts between music pieces in Equation (1) as:

$$\omega_{h,m} = \left\{ \begin{array}{l} \alpha^l_h f\left(\mathbf{v}_h, \mathbf{v}_m\right), \ if\ h \in H_u \backslash m_n \\ \alpha^s_h f\left(\mathbf{v}_h, \mathbf{v}_m\right), \ if\ h = m_n \end{array} \right. , \quad (7)$$

where $f(\cdot) : \mathcal{R}^d \times \mathcal{R}^d \to \mathcal{R}$ is a mapping function, which will be explained later.

C. Sequential Music Recommendation

Note that $f(\cdot)$ in Equation (7) can be defined as cosine similarity, dot product, or negative Euclidean distance. However, these metrics can only model correlations of the same dimension in embeddings. Inspired by Factorizing Personalized Markov Chains (FPMC) [27], we adopt a transition matrix to capture the inter-dimension correlations in the embeddings of music pieces. Therefore, $f(\cdot)$ can be formally defined as

$$f\left(\mathbf{v}_1, \mathbf{v}_2\right) = -\left\| \mathbf{A}\mathbf{v}_1 - \mathbf{v}_2 \right\|_2^2, \quad (8)$$

where $\mathbf{A} \in \mathcal{R}^d \times d$ is the transition matrix.

Then, we can use Equation (1) to calculate the preference of $u \in U$ for target item $m \in M$ at time t given H_u. However, the result of intensity function $\tilde{\lambda}_{m|u}(t)$ could be negative. Therefore, we use a softmax unit to define the probability that user u is interested in target music piece m at time t as

$$p_{m|u}(t) = \frac{\exp\left(\tilde{\lambda}_{m|u}(t)\right)}{\sum\limits_{k \in M} \exp\left(\tilde{\lambda}_{k|u}(t)\right)}. \quad (9)$$

For each target music piece $m \in M$, Equation (9) defines a conditional distribution $p_{\cdot|u}(t)$ over the entire music set M.

At last, we can perform recommendation according to the ranking scores of two item m and m', which is defined as

$$m >_{u,t} m' :\Leftrightarrow p_{m|u}(t) > p_{m'|u}(t). \quad (10)$$

Note that we take both next and next new music recommendation tasks into consideration in this work. Especially, the candidates in next music recommendation task are the whole music set, while the music pieces that have not been listened to by the target user yet are used as candidates in next new music recommendation task.

D. Model Learning

In the learning process, Equation (9) is maximized over all users' music listening sequences in the dataset. However, the complexity of softmax function in Equation (9) is proportional to the music set size $|M|$, which may reach millions in real-world online music services or applications. Therefore, we use a computationally efficient strategy, negative sampling [56] to approximate the original softmax function in Equation (9). Then, the log probability can be calculated approximately as:

$$\begin{aligned} \log p_{m|u}(t) \propto \log \sigma\left(\tilde{\lambda}_{m|u}(t)\right) \\ +n \cdot \mathbb{E}_{m' \sim P_M}\left[\log \sigma\left(-\tilde{\lambda}_{m'|u}(t)\right)\right], \end{aligned} \tag{11}$$

where $\sigma(x)$ is a *sigmoid* function, n is the count of "negative" samples, and m' is the music piece sampled from music set based on P_M, which is a noise distribution defined with empirical uni-gram distribution over music pieces. Note that n is much smaller than music set size $|M|$, so the training time is independent of the item set size $|M|$. Then, traditional optimization methods, such as stochastic gradient descent algorithms, can be adopted to optimize the objective function defined in Equation (11).

V. Experiments

In this section, extensive experiments are designed to answer the following research questions:

RQ1: Does ATPP outperform state-of-the-art baselines in next and next new music recommendation tasks?

RQ2: What are the effects of the three key components in the ATPP architecture?

RQ3: How does the dimension of embedding in ATPP affect the recommendation results?

A. Experimental Designs

The detailed experimental designs, including datasets, recommendation tasks, baselines and evaluation metrics, are introduced in this section.

*1) **Datasets**:* We evaluate the proposed approach ATPP and baselines on four real-world music datasets, including Xiami[3] [4], Lastfm[4] [12], 30music[5] [57] and LFM-1b[6] [58]. the statistics information of all datasets are listed in Table II. Note that sparsity means how sparse the user-music interaction data is. Specifically, if there exist k interaction records between m users and n music pieces, then the corresponding data sparsity is $1 - \frac{k}{m \times n}$. Moreover, Figure 4 gives popularity information (logarithm) of four music datasets mentioned above, which are consistent with the power law distribution [59].

Each dataset is split into a training set for training recommendation models and a test set for evaluation, which are non-overlapping. Specifically, we randomly choose 20% users from the dataset as test users and the rest 80% as train users. Then, the full music listening sequences of train users and the first half of the test users' sequences are used as train set, while the second half of test users' music listening sequences are used as test set. In particular, each user u's music listening sequence $H_u := \left[(u_1, m_1, t_1), (u_2, m_2, t_2), \ldots, (u_{|H_u|-1}, m_{|H_u|-1}, t_{|H_u|-1})\right]$ in the test set generates $|H_u| - 1$ test cases, where the k-th test case is to perform recommendation at time t_{k+1} given u's historical sequences $H_u := \left[(u_1, m_1, t_1), (u_2, m_2, t_2), \ldots, (u_k, m_k, t_k)\right]$ with the ground truth m_{k+1}.

*2) **Tasks**:* The tasks in this work include both next and next new music recommendation tasks. Specifically, the candidates in next music recommendation task are the whole music set, while the music pieces that have not been listened to by the target user yet are used as candidates in next new music recommendation task, which is more challenging and important. Especially, these two tasks can evaluate the recommendation methods' ability in exploiting and exploring users' preferences, which is a classic problem in recommendation research. In this work, we evaluate the proposed ATPP and baselines on both kinds of tasks, i.e., next music recommendation and next new music recommendation. Note that there is only one correct answer for one test case in both tasks.

*3) **Baselines**:* The following basic methods and state-of-the-art models are used as baselines:

- **Pop** performs recommendation based on items' popularity in training data.
- **PPop** (Personalized Pop) performs recommendation based on items' popularity for each user, which cannot perform next new music recommendation.
- **FPMC** [27] combines matrix factorization model with first-order Markov chain for sequential recommendation.
- **HRM** [28] encodes sequential patterns and users' general taste as one vector with hierarchical representation learning model. HRM-max and HRM-avg are two variants with max and average pooling aggregation, separately.
- **RDR** [4] can learn the feature vectors of items from behavior sequences with a skip-gram model [56], and acquire users' preferences for sequential recommendation.
- **SHAN** [11] uses a hierarchical attention mechanism to mine users' long/short-term preferences.
- **TiSASRec** [54] is a time interval aware self-attention based approach for sequential recommendation.
- **Mult-VAE** [60] is a collaborative fltering recommendation method for incorporating implicit feedback based on variational autoencoders.
- **SASRec** [35] models users' longer-term semantics as well as their recent actions simultaneously for accurate next item recommendation
- **Caser** [33] embeds items in users' behavior sequences into an "image" in the time and latent spaces and learns user preferences and sequential patterns for recommendation.
- **HGN** [36] adopts a feature gating module and an instance gating module to select informative latent features and items for sequential recommendation.

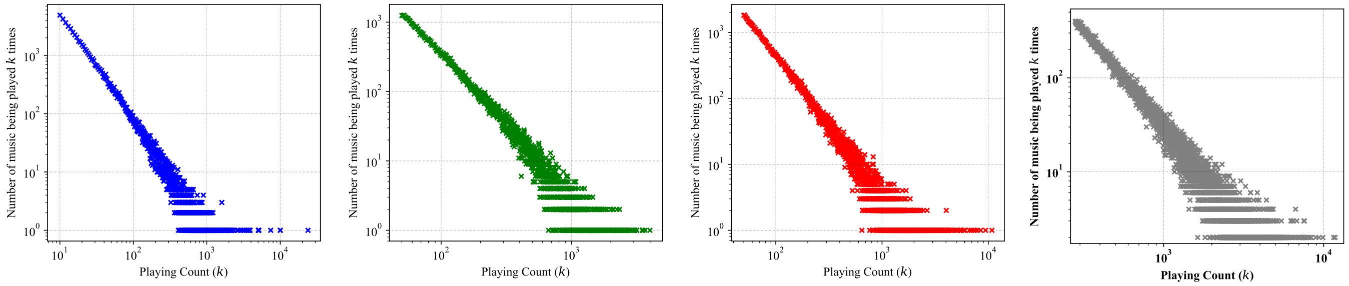
*4) **Evaluation Metrics**:* In the evaluation step, every method generates a recommendation list of k music pieces (top-k recommendation), which is evaluated by two quality metrics,

[3] https://1drv.ms/f/s!ApojZBGe9UzXgaI6x8pBf8JgN4PfZg

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

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

[6] http://www.cp.jku.at/datasets/LFM-1b/



(a) Xiami dataset (b) Lastfm dataset (c) 30music dataset (d) LFM-1b dataset

Figure 4. Popularity analysis of four music datasets

Table II
COMPLETE STATISTICS OF FOUR MUSIC DATASETS

Dataset	#Users	#Music Pieces	#Listening Records	#Avg.Records / User	#Avg.Records / Music	Sparsity
Xiami	3,982	64,334	3,154,815	792	49	98.77%
Lastfm	896	66,407	1,264,137	1,411	19	97.88%
30music	2,970	84,882	3,168,916	1,067	37	98.74%
LFM-1b	7,641	102,416	10,816,752	1,416	106	98.62%

i.e. recall and Mean Reciprocal Rank (MRR). Note that there is only one correct answer for each testcase in both tasks.

Recall is the fraction of the total amount of hits in all testcases. Specifically, a hit means the target music piece (ground truth) appears in the recommendation list. For instance, if there exists a listening record (u, m, t) in the test set and the recommended list of u contains m, then it is called a hit. Recall is formally defined as:

$$Recall@k = \frac{\#hit}{\#testcase}, \tag{12}$$

where k is the length of a recommendation list, $\#hit$ is the amount of hits, and $\#testcase$ is the amount of all testcases.

MRR is a ranking evaluation metric, which calculates the average of the reciprocal ranks of target music piece in a recommendation list, i.e.,

$$MRR@k = \frac{\sum 1/rank_k}{\#testcase}, \tag{13}$$

where $rank_n$ is the ranking of the n_{th} test case's target music piece in the generated recommendation list.

*5) **Implementation Details:*** In the training phase, we set the batch size to 512, negative sample size to 10, dimension of embedding to 128, number of epochs to 100. Besides, the parameters in model are updated via Adam optimizer [61] with the learning rate $3e-4$. Moreover, to prevent over-fitting, we set the weight decay in Adam as 0.01. All the experiments were implemented using the PyTorch 1.5.0 framework with Python 3.6, and the experiments were conducted on a server with 1.80 GHz Intel(R) Xeon(R) Silver 4108 CPU, an GeForce RTX 2080Ti GPU with 48 GB memory, running Ubuntu 18.04. The source code of ATPP is avalable on github[7].

B. Comparison with Baselines (RQ1)

To verify the effectiveness of our proposed approach, we compare ATPP with several state-of-the-art baselines on two tasks (i.e., next and next new music recommendation), over four dataset. The experimental results of next and next new music recommendation tasks are reported in Table III and Table IV, respectively. Note that, the numbers in bold font are the best results, and the second best results are underlined.

We can observe that our ATPP indeed outperforms other baselines in most of evaluation settings, and it can balance next and next new recommendation tasks. We attribute this to the fact that ATPP can learn the listening patterns/correlations in music playing sequences to recommend new music pieces that they may be interested in. Besides, ATPP can capture and leverage users' long/short-term user preferences in an effective and adaptive way. Especially, the performance on next new music recommendation show that ATPP can help users to explore interesting new music pieces, while the performance on next music recommendation show ATPP's ability of accurately predicting users's next listening behavior. Most baselines have limited support for next new music recommendation, because they rely on historical listening records.

In detail, ATPP achieves overall better performance than deep-learning based baselines. This is because that ATPP can fully exploit complex sequential patterns and temporal context in users' music listening sequences with a multi-variant temporal point process and learn the embeddings effectively from users' behavior sequences. Especially, the attention mechanism and transition matrix enhance ATPP's ability of inferring users' long/short-term preferences adaptively, and enable it to capture the temporal and context information that is essential for improving next and next new music recommendation. Besides, compared with ATPP, RDR and SASRec have better results in next music recommendation tasks

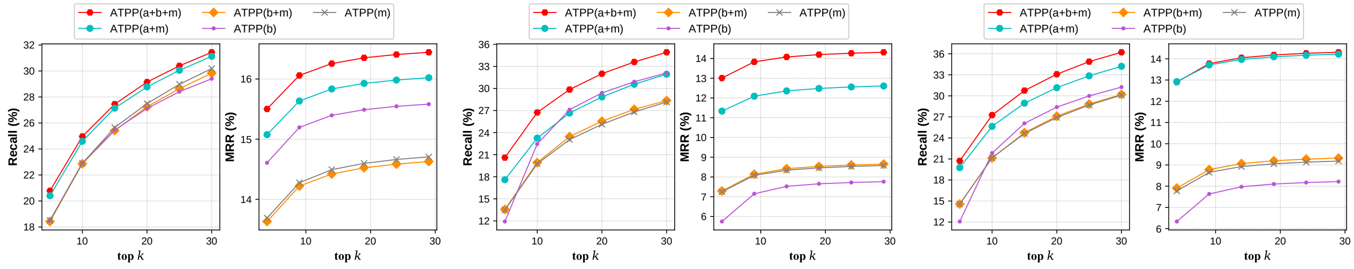
[7] https://github.com/ctokyo/ATPP

Table III
COMPARISONS BETWEEN THE PROPOSED APPROACH ATPP AND BASELINES ON **NEXT** MUSIC RECOMMENDATION TASK

Method	Xiami (%)				Lastfm (%)				30music (%)				LFM-1b (%)			
	R@10	MRR@10	R@20	MRR@20	R@10	MRR@10	R@20	MRR@20	R@10	MRR@10	R@20	MRR@20	R@10	MRR@10	R@20	MRR@20
Pop	2.300	1.303	3.013	1.354	0.295	0.097	0.543	0.114	0.441	0.155	0.671	0.170	0.346	0.116	0.602	0.133
PPop	16.140	7.565	22.871	8.026	6.238	2.477	9.683	2.711	4.257	1.481	6.888	1.661	7.327	2.807	11.499	3.092
HRM-max	8.077	2.838	11.858	3.100	4.614	1.312	7.708	1.520	4.672	1.421	8.162	1.660	8.174	2.448	14.264	2.864
HRM-avg	6.493	2.413	10.314	2.679	7.405	2.334	11.693	2.627	5.434	1.796	9.055	2.043	7.253	2.249	12.198	2.587
TiSASRec	16.936	12.565	19.447	12.740	9.083	4.523	12.678	4.772	6.678	3.569	9.306	3.750	13.563	6.499	18.430	6.834
FPMC	10.392	5.690	13.752	5.922	5.855	2.085	8.476	2.268	7.271	2.572	10.920	2.821	13.363	4.845	20.339	5.403
SHAN	18.013	10.173	22.434	10.477	14.685	5.332	21.832	5.825	12.251	4.632	18.473	5.059	8.733	2.999	13.381	3.323
RDR	**27.643**	**17.880**	31.062	**18.121**	24.559	8.220	31.058	8.680	23.825	8.375	29.899	8.802	25.354	9.174	33.195	9.726
Mult-VAE	3.643	1.153	6.521	1.346	1.9456	0.614	3.630	0.728	1.517	0.476	2.726	0.558	1.747	0.531	3.214	0.630
SASRec	25.970	14.879	**32.137**	15.315	20.114	6.173	28.923	6.800	20.036	6.258	28.820	6.885	23.582	7.750	34.615	8.537
Caser	11.237	6.598	14.273	6.804	10.589	3.939	15.694	4.291	5.912	2.207	9.103	2.426	5.007	1.780	7.833	1.974
HGN	15.442	8.994	19.896	9.298	7.573	2.630	11.871	2.924	7.221	2.625	11.457	2.916	9.454	3.307	15.331	3.708
ATPP	24.963	16.061	29.147	16.353	**26.737**	**13.831**	**32.002**	**14.198**	**27.253**	**13.772**	**33.082**	**14.180**	**34.897**	**16.383**	**42.888**	**16.944**

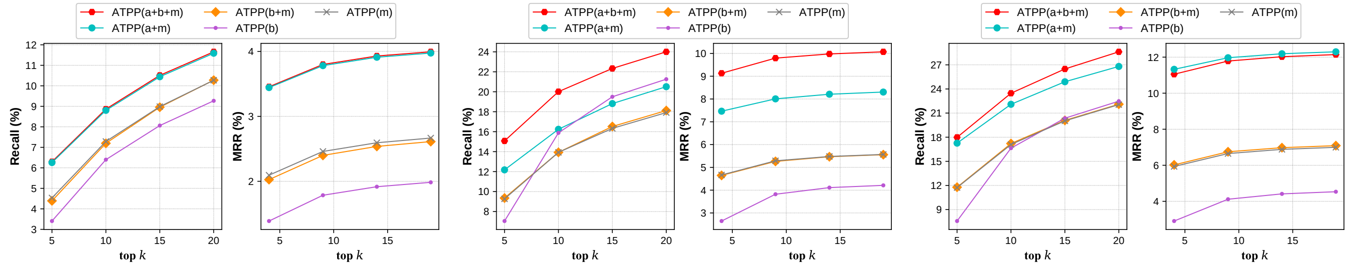
Table IV
COMPARISONS BETWEEN ATPP AND BASELINES ON **NEXT NEW** MUSIC RECOMMENDATION TASK

Method	Xiami (%)				Lastfm (%)				30music (%)				LFM-1b (%)			
	R@10	MRR@10	R@20	MRR@20	R@10	MRR@10	R@20	MRR@20	R@10	MRR@10	R@20	MRR@20	R@10	MRR@10	R@20	MRR@20
Pop	0.344	0.098	0.700	0.123	0.183	0.059	0.330	0.069	0.203	0.068	0.339	0.077	0.097	0.031	0.187	0.038
PPop	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
HRM-max	1.389	0.393	2.630	0.477	1.757	0.478	3.130	0.571	2.440	0.629	4.813	0.790	4.409	1.121	8.681	1.411
HRM-avg	1.534	0.458	2.735	0.539	2.348	0.725	4.260	0.855	2.408	0.728	4.376	0.861	3.272	0.907	6.298	1.112
FPMC	2.726	1.028	3.967	1.113	5.524	2.046	7.571	2.190	7.202	2.817	9.985	3.010	9.404	4.258	13.847	4.281
TiSASRec	1.883	0.501	3.145	0.586	3.440	0.870	5.724	1.026	2.482	0.633	4.419	0.764	7.627	1.905	11.927	2.200
SHAN	4.126	1.196	7.036	1.394	8.010	2.255	13.333	2.619	7.634	2.185	12.872	2.542	5.124	1.372	8.759	1.626
RDR	6.562	1.439	9.109	1.616	16.395	3.554	21.910	3.943	18.458	4.034	24.237	4.440	18.167	3.620	25.549	4.138
Mult-VAE	0.845	0.253	1.576	0.302	2.079	0.624	3.770	0.738	0.586	0.177	1.105	0.212	1.631	0.496	2.997	0.589
SASRec	5.793	1.024	9.801	1.305	10.933	1.753	18.345	2.280	12.373	1.955	20.481	2.534	14.251	2.422	24.705	3.162
Caser	2.869	0.883	4.683	1.005	6.668	2.214	10.363	2.466	4.268	1.438	6.859	1.614	3.488	1.202	5.663	1.350
HGN	2.719	0.776	4.869	0.921	2.267	0.692	4.050	0.813	3.407	1.000	6.198	1.189	4.825	1.298	9.153	1.592
ATPP	**8.864**	**3.799**	**11.658**	**3.994**	**20.009**	**9.794**	**24.001**	**10.070**	**23.468**	**11.784**	**28.620**	**12.144**	**30.864**	**13.739**	**38.241**	**14.254**



(a) Xiami dataset (b) Lastfm dataset (c) 30music dataset

Figure 5. Experimental results of ATPP's components on next music recommendation task. "a", "b", and "m" represent three key components in ATPP, which are attention, time-aware behavior, and transition matrix, separately.



(a) Xiami dataset (b) Lastfm dataset (c) 30music dataset

Figure 6. Experimental results of ATPP's components on next new music recommendation task. "a", "b", and "m" represent three key components in ATPP, which are attention, time-aware behavior, and transition matrix, separately.

on Xiami dataset. The reason is that the original music playing records in Xiami dataset are only accurate to the minute, while the timestamps in the other three datasets that are accurate to seconds. Different time accuracy influences the modeling of time-aware behaviors in ATPP to some extent. Note that the proposed approach ATPP still outperforms RDR in all the rest cases, especially in the next new music recommendation task.

When compared with Pop, PPop and Multi-VAE that ignore sequential information, our ATPP still keeps a better performance. The results further confirm necessity of considering users' music listening sequences as well as temporal and context information for the tasks of next and next new music recommendation.

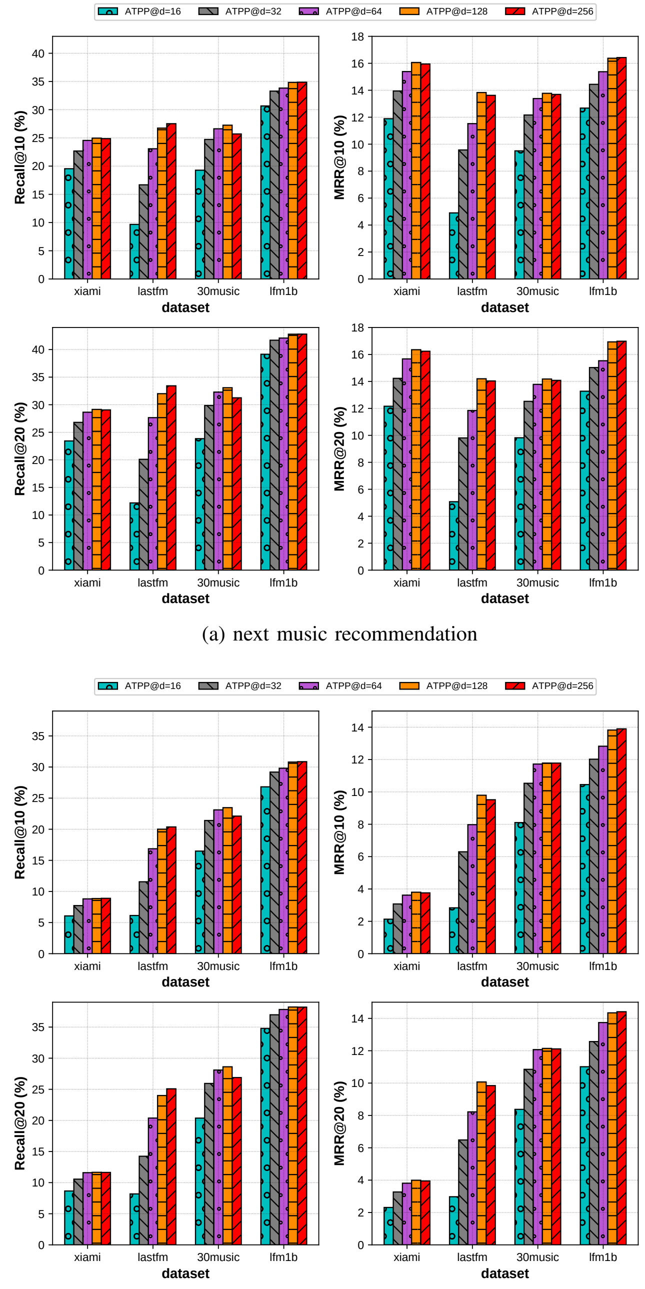
In conclusion, the comparison with baselines on four music datasets shows that the proposed approach ATPP is effective both next and next new music recommendation tasks.

C. *Effects of Key Components in* ATPP *(RQ2)*

We also investigate the effectiveness of three key components in the ATPP architecture via ablation analysis. Specifically, we evaluate some combinations of these components in ATPP, including ATPP (a+b+m), ATPP (a+m), ATPP (b+m), ATPP (b), ATPP (m). Note that "a", "b", and "m" represent three key components in ATPP, which are attention, time-aware behavior, and transition matrix, respectively. The results of next and next new music recommendation tasks on four datasets are given in Figure 5 and Figure 6, respectively. Note that Lastfm, 30music and LFM-1b dataset were collected from Last.fm, and we present the ablation experiments on Lastfm, Xiami and 30music dataset for simplicity.

We can observe that "ATPP (a+b+m)" that has all three components outperforms other variants overall in both next and next new music recommendation tasks on all datasets. The results show that all the three components (i.e., attention, time-aware behavior, and transition matrix), play important roles in improving performance of sequential music recommendation. Note that the "ATPP (a+m)" achieves better performance in metric of MRR when perform next new music recommendation task on 30music dataset. One reason is that the "b" component for incorporating temporal behaviors may guide ATPP to exploiting users' preferences instead of exploration.

Besides, "ATPP (b+m)", "ATPP (a+m)", "ATPP (b)", and "ATPP (m)" have different results in term of different metrics, and their performance also changes on different datasets. For example, ATPP (b+m) outperforms "ATPP (b)" and "ATPP (m)" on Lastfm and 30music datasets in term of MRR, but its performance is not as good as the results of two variants in other cases. The reason is two-fold. Firstly, different datasets have different properties, which may influence the performance of recommendation methods. For example, the timestamps in Xiami datasets are in minutes while the records in the other datasets are accurate to second. Therefore, the time-aware behavior component ("b") on Xiami dataset is not as effective as it on other datasets. Secondly, Recall is used to measure whether the recommendation method returns the relevant results, while MRR focuses on the ranking of relevant items. In other words, those two metrics evaluate the performance of recommendation methods from different perspectives.



(a) next music recommendation

(b) next new music recommendation

Figure 7. Experimental results of the dimension's impact

Overall, the results show that ATPP can effectively combine three key components to capture time and sequential information as well as users' long/short-term preferences, and leverage both preferences adaptively in sequence modeling and next (new) music recommendation.

D. *Impact of Embedding's Dimension (RQ3)*

Embeddings with higher dimension can capture/represent more useful information at the cost of higher time/space complexity and more computation resources. We evaluate the proposed model ATPP with different dimensions (16, 32, 64, 128 and 256) to investigate the impact of embedding's dimension on recommendation performance and determine

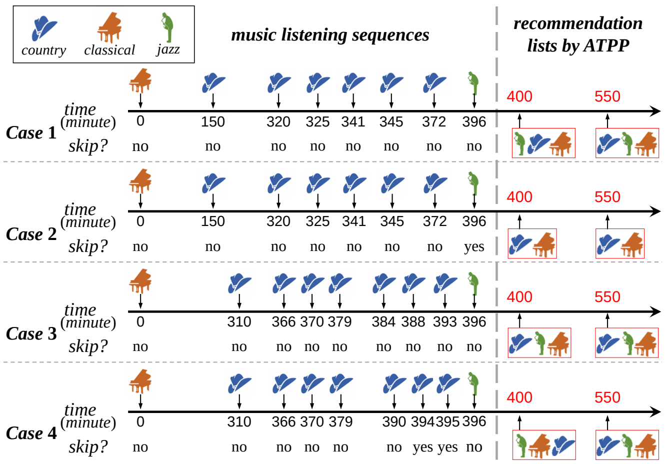


Figure 8. A case study for ATPP. The recommendation lists by ATPP vary with the listening sequences and the corresponding target time (red numbers).

the proper dimension to achieve good performance with comparatively low complexity. As shown in Figure 7a and Figure 7b, ATPP with higher embedding dimension has better performance in both metrics, i.e., recall and MRR, which shows that high-dimensional embeddings can indeed capture useful features and model users and music pieces accurately. Besides, the accuracy tends to get relatively stable and even drops when the dimension reaches 256. The reason is that higher dimension may result in over-fitting. Therefore, we set the embeddings' dimension as 128 for other experiments.

E. Case Study and Analysis

As shown in Figure 8, four case studies are designed to illustrate characteristics of ATPP. Firstly, ***case* 1** shows that different target time (400 and 550) will yield different recommendation results. Specifically, music recommendation results depend on user's previous listening behaviors, and the historical record at 396 (jazz music) has more influence on the prediction at 400. Besides, the recommendation lists at 550 depend more on users' listening records for country music, which are still in the majority. Secondly, ***case* 2** illustrates the influence of skip behaviors. More precisely, user skipped jazz music at 396, which indicates that he/she is probably not interested in jazz music, and the recommendation results at 400 and 550 are more likely to be country music. Thirdly, in ***case* 3**, user's most recent record for jazz music at 396 is not a skip behavior. However, most of her/his recent behaviors are country music, so country music ranks first in the recommendation lists generated by ATPP at 400 and 550. Fourthly, as shown in ***case* 4**, user has listened to a lot of country music recently, so the recommendation result at 550 are most likely to be country music. Furthermore, he/she skipped two piece of country music and then finished listening to the jazz music. In other words, ATPP infers that the user prefers jazz music to country music at 400. In conclusion, ATPP can exploit music listening sequences and temporal information, and leverage users' long/short-term preferences for accurate sequential music recommendation.

Besides, according to our studies, users' short-term preferences play a more important role than long-term preferences in most recommendation cases. The reason is three-fold. Firstly, users's long-term preferences may be diverse and various, but they usually prefer only one or a few kinds of music under a certain context (short-term preferences). For example, a user, who likes both light music and rock music, may prefer the latter when working out. Therefore, users' short-term preferences contribute more to the sequential recommendation tasks, including both next and next new recommendation, which is different with traditional top-n recommendation. Secondly, the music listening is a kind of typical sequential behaviors, since users may listen to many music pieces continuously, and there are strong correlations/patterns between records in sequences, especially nearby ones. Note that the influence of long/short-term preferences on the prediction of next and next new music depends on both tempral information, such as time interval, and behavior information, such as skip behavior.

VI. Conclusion and Future Work

In this work, we have proposed a novel sequential recommendation method named Attentive Temporal Point Process (ATPP), which combines temporal point process and attention mechanism for sequentia music recommendation. It's worthy to highlight some advantages of our proposed approach when compared with existing approaches. Our ATPP can: 1) effectively exploit complex music listening sequences with temporal context; 2) accurately model dynamic relevance and complex relationships between music pieces in listening sequences; and 3) adaptively incorporate and leverage users' long/short-term user preferences for sequential music recommendation. Comprehensive experiments on four real-world music datasets verify the effectiveness of ATPP in both next and next new music recommendation tasks.

In future, we will explore how to combine TPP model with advanced sequence models, such as Transformer [62], for further improving the recommendation performance. As for cold start and data sparsity problem, we plan to incorporate more auxiliary/side information, such as users' social data [6], items' content features and heterogeneous information [63]. Besides, we will also try to further alleviate the cold start problem via meta-optimization idea [64], which can learn the user preference by only a few of past interacted items.

Acknowledgements

This work was supported by Natural Science Foundation of Zhejiang Province under No.LQ20F020015 and No.LQ21F020015, the Fundamental Research Funds for the Provincial University of Zhejiang under No.GK199900299012-017, and the Key Science and Technology Project of Zhejiang Province under No.2020C01165.

References

[1] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender systems handbook*. Springer, 2011, pp. 1–35.

[2] Y. Yang, Y. Xu, E. Wang, J. Han, and Z. Yu, "Improving existing collaborative filtering recommendations via serendipity-based algorithm," *IEEE Transactions on Multimedia*, vol. 20, no. 7, pp. 1888–1900, 2018.

[3] L. Yang, D. Wu, Y. Cai, X. Shi, and Y. Wu, "Learning-based user clustering and link allocation for content recommendation based on d2d multicast communications," *IEEE Transactions on Multimedia*, vol. 22, no. 8, pp. 2111–2125, 2020.

[4] D. Wang, S. Deng, and G. Xu, "Sequence-based context-aware music recommendation," *Information Retrieval Journal*, vol. 21, no. 2-3, pp. 230–252, 2018.

[5] T. K. Paradarami, N. D. Bastian, and J. L. Wightman, "A hybrid recommender system using artificial neural networks," *Expert Systems with Applications*, vol. 83, pp. 300–313, 2017.

[6] Z. Zhao, Q. Yang, H. Lu, T. Weninger, D. Cai, X. He, and Y. Zhuang, "Social-aware movie recommendation via multimodal network learning," *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 430–440, 2018.

[7] Y. Zhou, J. Wu, T. H. Chan, S.-W. Ho, D.-M. Chiu, and D. Wu, "Interpreting video recommendation mechanisms by mining view count traces," *IEEE Transactions on Multimedia*, vol. 20, no. 8, pp. 2153–2165, 2018.

[8] S. Jiang, X. Qian, J. Shen, Y. Fu, and T. Mei, "Author topic model-based collaborative filtering for personalized poi recommendations," *IEEE transactions on multimedia*, vol. 17, no. 6, pp. 907–918, 2015.

[9] M. Schedl, P. Knees, B. McFee, D. Bogdanov, and M. Kaminskas, "Music recommender systems," in *Recommender systems handbook*. Springer, 2015, pp. 453–492.

[10] M. Quadrana, P. Cremonesi, and D. Jannach, "Sequence-aware recommender systems," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, pp. 1–36, 2018.

[11] H. Ying, F. Zhuang, F. Zhang, Y. Liu, G. Xu, X. Xie, H. Xiong, and J. Wu, "Sequential recommender system based on hierarchical attention network," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, ser. IJCAI'18. AAAI Press, 2018, p. 3926–3932.

[12] O. Celma, "Music recommendation," in *Music recommendation and discovery*. Springer, 2010, pp. 43–85.

[13] E. Shakirova, "Collaborative filtering for music recommender system," in *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. IEEE, 2017, pp. 548–550.

[14] N. Sachdeva, K. Gupta, and V. Pudi, "Attentive neural architecture incorporating song features for music recommendation," in *Proceedings of the 12th ACM Conference on Recommender Systems*, 2018, pp. 417–421.

[15] E. Zheng, G. Y. Kondo, S. Zilora, and Q. Yu, "Tag-aware dynamic music recommendation," *Expert Systems with Applications*, vol. 106, pp. 244–251, 2018.

[16] D. Cheng, T. Joachims, and D. Turnbull, "Exploring acoustic similarity for novel music recommendation," in *Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020*, 2020, pp. 583–589.

[17] T. Shen, J. Jia, Y. Li, Y. Ma, Y. Bu, H. Wang, B. Chen, T.-S. Chua, and W. Hall, "Peia: Personality and emotion integrated attentive model for music recommendation on social media platforms," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 206–213.

[18] H.-T. Zheng, J.-Y. Chen, N. Liang, A. K. Sangaiah, Y. Jiang, and C.-Z. Zhao, "A deep temporal neural music recommendation model utilizing music and user metadata," *Applied Sciences*, vol. 9, no. 4, p. 703, 2019.

[19] Z. Cheng and J. Shen, "On effective location-aware music recommendation," *ACM Transactions on Information Systems (TOIS)*, vol. 34, no. 2, pp. 1–32, 2016.

[20] S. Deng, D. Wang, X. Li, and G. Xu, "Exploring user emotion in microblogs for music recommendation," *Expert Systems with Applications*, vol. 42, no. 23, pp. 9284–9293, 2015.

[21] D. Ayata, Y. Yaslan, and M. E. Kamasak, "Emotion based music recommendation system using wearable physiological sensors," *IEEE transactions on consumer electronics*, vol. 64, no. 2, pp. 196–203, 2018.

[22] E. Zangerle, C.-M. Chen, M.-F. Tsai, and Y.-H. Yang, "Leveraging affective hashtags for ranking music recommendations," *IEEE Transactions on Affective Computing*, vol. 12, no. 1, pp. 78–91, 2018.

[23] A. Vall and G. Widmer, "Machine learning approaches to hybrid music recommender systems," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2018, pp. 639–642.

[24] D. Wang, S. Deng, X. Zhang, and G. Xu, "Learning to embed music and metadata for context-aware music recommendation," *World Wide Web*, vol. 21, no. 5, pp. 1399–1423, 2018.

[25] J. H. Lee, L. Pritchard, and C. Hubbles, "Can we listen to it together?: Factors influencing reception of music recommendations and post-recommendation behavior," in *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019*, 2019, pp. 663–669.

[26] B. Manolovitz and M. Ogihara, "Practical evaluation of repeated recommendations in personalized music discovery," in *Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020*, 2020, pp. 633–639.

[27] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 811–820.

[28] P. Wang, J. Guo, Y. Lan, J. Xu, S. Wan, and X. Cheng, "Learning hierarchical representation model for nextbasket recommendation," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2015, pp. 403–412.

[29] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *4th International Conference on Learning Representations, ICLR 2016*, 2016.

[30] Y. Zhu, H. Li, Y. Liao, B. Wang, Z. Guan, H. Liu, and D. Cai, "What to do next: Modeling user behaviors by time-lstm." in *IJCAI*, vol. 17, 2017, pp. 3602–3608.

[31] P. Zhao, A. Luo, Y. Liu, F. Zhuang, J. Xu, Z. Li, V. S. Sheng, and X. Zhou, "Where to go next: A spatio-temporal gated network for next poi recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[32] A. Vall, M. Dorfer, H. Eghbal-Zadeh, M. Schedl, K. Burjorjee, and G. Widmer, "Feature-combination hybrid recommender systems for automated music playlist continuation," *User Modeling and User-Adapted Interaction*, vol. 29, no. 2, pp. 527–572, 2019.

[33] J. Tang and K. Wang, "Personalized top-n sequential recommendation via convolutional sequence embedding," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 2018, pp. 565–573.

[34] F. Yuan, A. Karatzoglou, I. Arapakis, J. M. Jose, and X. He, "A simple convolutional generative network for next item recommendation," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019, pp. 582–590.

[35] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2018, pp. 197–206.

[36] C. Ma, P. Kang, and X. Liu, "Hierarchical gating networks for sequential recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 825–833.

[37] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 1441–1450.

[38] A. G. Hawkes, "Spectra of some self-exciting and mutually exciting point processes," *Biometrika*, vol. 58, no. 1, pp. 83–90, 1971.

[39] R. Rotondi and E. Varini, "Failure models driven by a self-correcting point process in earthquake occurrence modeling," *Stochastic environmental research and risk assessment*, vol. 33, no. 3, pp. 709–724, 2019.

[40] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song, "Recurrent marked temporal point processes: Embedding event history to vector," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1555–1564.

[41] H. Mei and J. M. Eisner, "The neural hawkes process: A neurally self-modulating multivariate point process," in *Advances in Neural Information Processing Systems*, 2017, pp. 6754–6764.

[42] J. Zhang, Z. Wei, Z. Yan, M. Zhou, and A. Pani, "Online change-point detection in sparse time series with application to online advertising," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 6, pp. 1141–1151, 2019.

[43] H. Xu, W. Wu, S. Nemati, and H. Zha, "Patient flow prediction via discriminative learning of mutually-correcting processes," *IEEE transactions on Knowledge and Data Engineering*, vol. 29, no. 1, pp. 157–171, 2017.

[44] H. S. Dutta, V. R. Dutta, A. Adhikary, and T. Chakraborty, "Hawkeseye: Detecting fake retweeters using hawkes process and topic modeling," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2667–2678, 2020.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[46] W. Zhang, S. Tang, Y. Cao, S. Pu, F. Wu, and Y. Zhuang, “Frame augmented alternating attention network for video question answering,” *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 1032–1041, 2020.

[47] J. Song, J. Xiao, F. Wu, H. Wu, T. Zhang, Z. M. Zhang, and W. Zhu, “Hierarchical contextual attention recurrent neural network for map query suggestion,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 9, pp. 1888–1901, 2017.

[48] X. Gao, F. Feng, X. He, H. Huang, X. Guan, C. Feng, Z. Ming, and T.-S. Chua, “Hierarchical attention network for visually-aware food recommendation,” *IEEE Transactions on Multimedia*, vol. 22, no. 6, pp. 1647–1659, 2020.

[49] D. Wang, X. Zhang, D. Yu, G. Xu, and S. Deng, “Came: Content-and context-aware music embedding for recommendation,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2020.

[50] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, “Neural attentive session-based recommendation,” in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1419–1428.

[51] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, “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*. ACM, 2017, pp. 335–344.

[52] J. Xiao, H. Ye, X. He, H. Zhang, F. Wu, and T.-S. Chua, “Attentional factorization machines: Learning the weight of feature interactions via attention networks,” in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, ser. IJCAI’17. AAAI Press, 2017, pp. 3119–3125.

[53] J. Han, L. Zheng, Y. Xu, B. Zhang, F. Zhuang, S. Y. Philip, and W. Zuo, “Adaptive deep modeling of users and items using side information for recommendation,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 3, pp. 737–748, 2020.

[54] J. Li, Y. Wang, and J. McAuley, “Time interval aware self-attention for sequential recommendation,” in *Proceedings of the 13th International Conference on Web Search and Data Mining*, 2020, pp. 322–330.

[55] X. Luo, M. Zhou, S. Li, D. Wu, Z. Liu, and M. Shang, “Algorithms of unconstrained non-negative latent factor analysis for recommender systems,” *IEEE Transactions on Big Data*, pp. 1–1, 2019.

[56] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[57] R. Turrin, M. Quadrana, A. Condorelli, R. Pagano, and P. Cremonesi, “30music listening and playlists dataset.” in *Poster Proceedings of the 9th ACM Conference on Recommender Systems*, 2015.

[58] M. Schedl, “The lfm-1b dataset for music retrieval and recommendation,” in *Proceedings of the 2016 ACM on international conference on multimedia retrieval*, 2016, pp. 103–110.

[59] L. A. Adamic and B. A. Huberman, “Power-law distribution of the world wide web,” *Science*, vol. 287, no. 5461, pp. 2115–2115, 2000.

[60] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, “Variational autoencoders for collaborative filtering,” in *Proceedings of the 2018 world wide web conference*, 2018, pp. 689–698.

[61] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[62] Q. Zhang, A. Lipani, O. Kirnap, and E. Yilmaz, “Self-attentive hawkes process,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 11 183–11 193.

[63] S. Liu, I. Ounis, C. Macdonald, and Z. Meng, “A heterogeneous graph neural model for cold-start recommendation,” in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 2029–2032.

[64] M. Dong, F. Yuan, L. Yao, X. Xu, and L. Zhu, “Mamo: Memory-augmented meta-optimization for cold-start recommendation,” in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 688–697.

Dongjing Wang (Member, IEEE) received the B.S. and Ph.D. degrees in computer science from Zhejiang University, Hangzhou, China, in 2012 and 2018, respectively. He was co-trained at the University of Technology Sydney, Ultimo, NSW, Australia, for one year. He is currently a Lecturer with Hangzhou Dianzi University, Hangzhou. His current research interests include recommender systems, machine learning, data mining, and business process management.

Xin Zhang received the bachelor’s and Ph.D. degrees in computer science and technology from Shandong University, Jinan, China, in 2012 and 2018, respectively. She was co-trained at the University of California, Davis, CA, USA, for one year. She is currently a Lecturer with the School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China. Her research interests include deep learning, image processing and computer vision.

Yao Wan received his Ph.D. degree from the College of Computer Science, Zhejiang University, Hangzhou, China, in 2019. He is currently a Lecturer with the School of Computer Science and Technology, Huazhong University of Science and Technology. He has been a visiting student of University of Technology Sydney and University of Illinois at Chicago in 2016 and 2018, respectively. His research interests lie in the synergy between artificial intelligence and software engineering.

Dongjin Yu (Senior Member, IEEE) is currently a Professor with Hangzhou Dianzi University, Hangzhou, China, where he is also the Director of the Institute of Big Data and the Institute of Computer Software. His research efforts include big data, business process management, and software engineering. Dr. Yu is also a member of ACM and a Senior Member of the China Computer Federation (CCF). He is also a member of the Technical Committee of Software Engineering of CCF and the Technical Committee of Service Computing of CCF.

Guandong Xu (Member, IEEE) is a Full Professor in Data Science at School of Computer Science and Advanced Analytics Institute, University of Technology Sydney with PhD degree in Computer Science. His research interests cover Data Science, Data Analytics, Recommender Systems, Web Mining, User Modelling, Social Network Analysis, and Social Media Mining. He has published three monographs in Springer and CRC press, and 190+ journal and conference papers including TNNLS, TSC, IJCAI, AAAI, WWW, ICDE, and CVPR conferences. He is the assistant Editor-in-Chief of World Wide Web Journal and has been serving in editorial board or as guest editors for several international journals. He has received a number of Industry Awards from Australian industry community, such as 2018 Top-10 Australian Analytics Leader Award.

Shuiguang Deng (Senior Member, IEEE) received the B.S. and Ph.D. degrees in computer science from Zhejiang University, Hangzhou, China, in 2002 and 2007, respectively. He was a Visiting Scholar with the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 2014, and Stanford University, Stanford, CA, USA, in 2015. He is currently a Full Professor with the College of Computer Science and Technology, Zhejiang University. He has published over 80 articles in journals, including TOC, TPDS, TSC, TCYB, TNNLS, and refereed conferences. His research interests include service computing, mobile computing, and business process management.