COAT: A Music Recommendation Model based on Chord Progression and Attention Mechanisms

Weite Feng, Tong Li*, Zhen Yang
Beijing University of Technology
*litong@bjut.edu.cn

Abstract—**Recently, efforts have been made to explore introducing music content into deep learning-based music recommendation systems. In previous research, with reference to tasks such as speech recognition, music content is often fed into recommendation models as low-level audio features, such as the Mel-frequency cepstral coefficients. However, unlike tasks such as speech recognition, the audio of music often contains multiple sound sources. Hence, low-level time-domain-based or frequency-domain-based audio features may not represent the music content properly, limiting the recommendation algorithm's performance. To address this problem, we propose a music recommendation model based on chord progressions and attention mechanisms. In this model, music content is represented as chord progressions rather than low-level audio features. The model integrates user song interactions and chord sequences of music and uses an attention mechanism to differentiate the importance of different parts of the song. In this model, to make better use of the historical behavioral information of users, we refer to the design of the neural collaborative filtering algorithm to obtain embedding of users and songs. Under this basis, we designed a chord attention layer to mine users' fine-grained preferences for different parts of the music content. We conducted experiments with a subset of the last.fm-1b dataset. The experimental results demonstrate the effectiveness of the method proposed in this paper.**

Index Terms—**Data Mining, Recommendation System, Music Information Retrieval, Machine Learning**

I. Introduction

In recent years, with the growth of the mobile internet, access to music from internet music platforms has become convenient. Exposure to new music productions through recommendation algorithms is becoming a new way of consuming music. As such, music recommendation algorithms are broadly applied in the industry and have captured the interests of many academics.

Common music recommendation algorithms can be broadly classified into two categories, one known as Collaborative Filtering (CF) and one known as Content-Based Filtering (CBF) [1]. The CF method learns the user's preferences from the user's interaction with the song. The CBF approach makes a recommendation based on similarities that are calculated using song's labels or audio contents. Since audio content reflects music's content directly, establishing a relationship between users and audio content will help achieve accurate music recommendations.

To this end, efforts have been made to hybridize the user's listening history with the audio content to generate recommendations. For example, Oord et al. introduce Convolutional Neural Network (CNN) into the music recommendation task [2]. They first obtain a representation vector of music items via the CF method and then learns the mapping of audio content to the music vector by training a CNN to generate an embedding vector for new music. Lee et al. propose a user embedding approach, which integrates user history records with audio content in the framework of Neural Collaborative Filtering (NCF) [3] and generates recommendations end-to-end [4].

Due to the success of audio features in tasks such as speech recognition, audio content is often represented as frequency domain features such as Mel-frequency cepstral coefficients or spectrograms in hybrid music recommendation algorithms [5]. However, while these low-level features have been shown to be suitable to for certain tasks, their discriminative power and semantics are limited. This makes them may be unsuitable for music classification, musical emotion recognition, or music recommendation tasks, which require better representations [6].

At the same time, users tend to have different degrees of preference for different segments of music content, termed fine-grained music preferences [7]. However, existing embedding methods for music content do not distinguish between different parts of the music at a fine-grained level. Instead, use CNNs or Recurrent Neural Networks (RNNs) to directly learn the mapping between the audio content and the embedding vector. Such coarse-grained embedding methods may trap existing methods into sub-optimal solutions. Therefore, how to appropriately represent music content in a hybrid music recommendation system and tap into the fine-grained music preferences of users for music content becomes a problem that needs to be addressed.

For the representation of music content, we propose to use higher-level music features such as the chord progression instead of audio features. In musical compositions, a chord progression is a continuous sequence of chords (e.g., C-G-Am-F) that describes the structure of music, which is the defining feature on which melody and rhythm are built. The chord features showed better performance than the low-level audio features in the music emotion classification task [8]. For the mining of users' fine-grained music preferences, we propose

DOI reference number: 10.18293/SEKE2022-115

to leverage attention mechanisms to learn users' preferences for different parts of the music, which have recently been used as an effective means to fine-grained data mining [9], [10].

We argue that users' fine-grained preferences on music content should be carefully mined in order to render better recommendations. To this end, we proposed a music recommendation model based on chord progression and attention mechanisms (COAT), which combines user-item interaction with music content. In the COAT model, we use a Generalized Matrix Factorization (GMF) layer to mimic matrix factorization and mining users' musical interests from their interaction with songs. Based on this, we design a chord attention layer to calculate user attention scores for the different chords on the chord progression and generate music content embedding. Finally, we use a prediction layer to combine the output of the GMF layer and the chord attention layer to predict the listening probabilities. We conducted experiments with a subset of the last.fm-1b dataset to assess the performance of our proposal. The experimental results show that our approach outperforms the baseline.

II. Related Work

A. Music Recommendation Algorithm

Deep learning-based music recommendation approaches usually use deep learning techniques to obtain a vector representation of a song from its audio content or metadata, known as an embedding vector. The obtained embedding vectors are then used to perform content-based recommendations, integrated into matrix factorization methods, or build hybrid music recommendation systems [5].

Oord et al. introduce deep learning techniques to music recommendation systems [2]. After obtaining the embedding vectors of the users and songs by implementing a matrix factorization method, they train a CNN to learn the mapping between the audio features and its embedding vector. This allows new generated music to obtain its embedding vector via this CNN without interaction with the users. Beyond the audio content, scholars try to integrate information in more modalities. Yi et al. propose a cross-modal variable autoencoder for content-based micro-video background music recommendations that integrates video content and audio content to form recommendations [11].

Since separating the process of acquiring music embedding vectors from acquiring user and song embedding vectors may produce sub-optimal solutions, some scholars consider an end-to-end manner to build hybrid recommendation systems [4]. Liang et al. suggest a hybrid approach. The method first learns the content features through a multi-layer neural network and subsequently integrates them into the matrix factorization as a prior [12]. Lee et al. suggest a deep content-user embedding model, which learns user and song embedding through a multilayer neural network while using a CNN to learn the audio features of the song, and combines the two in an end-to-end way to finally generate recommendations [4]. Feng et al. proposed a hybrid music recommendation algorithm that combines user behavior and audio features to learn the fine-grained preferences of users for music content from multiple audio features by using an attention mechanism [7].

To sum up, much work has been done on integrating audio content into collaborative filtering recommender systems. However, these approaches have not yet explored the effects of higher-order music features with more explicit meanings in music recommendation tasks, nor have they been able to mine the fine-grained preferences of users for music content. The development of music information retrieval techniques and the application of attention mechanisms in recommender systems make it possible to fill this gap.

B. Attention-based Recommendation and Data Mining System

The human attention mechanism inspires the attention mechanism in deep learning. Like the attention to a specific part of the input in human vision, applying attention mechanisms in recommender systems allows the model to filter the most informative part from the input features. Hence reducing the influence of noisy data and thus improving the effectiveness of recommendations and bringing some interpretability [9], [13].

Wang et al. introduced the attention mechanism into the collaborative filtering method and proposed a dynamic user modeling approach based on the attention mechanism [14]. The method accurately portrays user interests by combining temporal information from calculating the degree of influence of the K items that the user has recently interacted with. The incorporation of the attention mechanism enhances the effectiveness of the CF method. Zhou et al. proposed a framework based on self-attention for modeling user behavior. The introduced self-attention mechanism demonstrated better performance and efficiency in their experiments than CNNs and RNNs [15].

Du et al. introduce an attention mechanism in the integration of user embedding vectors in a sarcasm detection task, which allows the features of various aspects of the user to be used effectively [16]. For the next point-of-interest recommendation task, Liu et al. proposed an attention-based category-aware GRU (ATCA-GRU) model [17]. The ATCA-GRU model can select the more significant parts of the relevant historical check-in trajectory to enhance the recommendation effect using the attention mechanism.

Gong et al. introduced an attention mechanism on the MOOC recommendation task. They fused meta-paths with contextual information by applying the attention mechanism on meta-paths of heterogeneous graphs to capture different students' different interests [18]. Shi et al. propose a method based on meta-paths and the attention mechanism [19]. Their proposed approach uses an attention mechanism to differentiate the importance of different meta-paths, which improves the effectiveness of recommendations and brings some interpretability.

In summary, attention mechanisms have been widely used with good results in various data mining tasks, which allows us to apply them to the analysis of users' fine-grained music preferences.

III. Method

The architecture of our proposed COAT model is shown in Figure 1. COAT model takes the one-hot vector of users and songs and the corresponding audio file of the song as input, and the output is the probability that the user listens to the song. Within the model, we model the history of user-song interactions in a neural collaborative filtering framework designed by He et al [3]. Above this, we design a chord attention layer to differentiate the importance of different parts of the song. After obtaining the vectors representing the user's long-term interests and the vectors representing the music content, we use a stacked neural network (called a prediction layer) to learn the complex relationship between user behavior records and music content and thus generate recommendations.

A. Generalized Matrix Factorization Layer

The user's interaction history is the most important representation of the user's interests, and mining the user's interest preferences can help improve the recommendation effect. Matrix factorization is a representative technique for this task, and being able to mimic this technique in a recommendation model is the foundation for building a successful recommendation model [3].

For this, we use a generalized matrix factorization layer (GMF) to mimic the matrix factorization. This layer first receives one-hot vector representations of the user and the song, denoted by V_u^U and V_i^I. After the embedding operation is implemented, these high-dimensional one-hot vectors are mapped to lower-dimensional vectors, called user and song embedding vectors. In the work of He et al., after obtaining the embedding vectors for the user and the song, the probability of the user clicking on the song can be obtained directly from the inner vector product. Here we keep these two vectors and input them into the next part of the model. The function is as follow:

$$\mathrm{GMF}_{out} = W_u^T V_u^U \odot W_i^T V_i^I \tag{1}$$

where W_u and W_i are the learnable parameters that map V_u^U and V_i^I into embedding vectors, and $\odot$ denotes the element-wise product of vectors.

B. Chord Attention Layer

The chord attention layer first takes in the audio file and then performs chord extraction. At this stage, we use the chord-extractor tool[1] to extract the chords of the music. As the number of chords is often inconsistent from song to song, and neural networks cannot handle variable-length data, we uniformly padded the collected chord sequences to 100 by repeating them in order.It is worth mentioning that as chord progressions are always repeated in a song, thus this treatment will not affect the data quality.

After obtaining the chord sequence, we generate a random embedding vector representation for each chord. Each chord sequence can be represented by a matrix C, and C_i represents the i-th chord vector in the chord matrix. The value of i indicates the order in which the chord appear in the time dimension. And since different placements of the same chords produce different sounds (e.g., Am-F-C-G and C-G-Am-F are different chord progressions), we generate positional embedding for each position, representing as p_i.

Combining the user embedding vector e_u and the position vector p_i, we calculate the attention weight a_i of each chord embedding vector using the following equation:

$$a_i = \mathrm{h}^{\mathrm{T}} Relu\left(\mathrm{W}^T \begin{bmatrix} \mathrm{e_u} \\ \mathrm{C_i} \\ \mathrm{p_i} \end{bmatrix} + \mathrm{b}\right) \tag{2}$$

Where h, W and b are parameters, and $Relu$ is the Relu activation function.

After applying equation 2 to calculate the individual attention weights a_i, we normalized them using softmax with the following equation:

$$\beta_{\mathrm{i}} = \frac{\exp\left(\mathrm{a_i}\right)}{\sum_{\mathrm{j}=1}^{|\mathrm{C}|} \exp\left(\mathrm{a_j}\right)} \tag{3}$$

The normalized attention weights β_i represent the importance of the different parts of the song, with which we finally weighted and summed with the chord embedding vector to obtain the output of the chord attention layer Att_{out}. The function is as follow:

$$Att_{\mathrm{out}} = \sum_{i=1}^{|C|} \beta_i \cdot C_i \tag{4}$$

C. Prediction layer

After feeding the model with information on chord sequences that represent the content of the music, the preference relationship between the user and the song becomes complex. Therefore, the model needs to have a stronger fitting capability to fit this kind of preference relationship.

Thus, after obtaining the vector GMF_{out} which contains information on user behavior and the vector Att_{out} which contains information on song content from the other two parts of the model, we calculate the final listening probability using a stacked neural network with the following equation:

$$\hat{\mathrm{y}}_{\mathrm{ui}} = MLP\left(\begin{bmatrix} GMF_{\mathrm{out}} \\ Att_{\mathrm{out}} \end{bmatrix}\right) \tag{5}$$

MLP stands for the common multi-layer perceptron, whose number of layers and shape can be set flexibly. In this paper, we set its number of layers to 3 to avoid too many parameters causing overfitting. In terms of shape, we set it using a typical tower structure, where each layer has twice the number of neurons as the next layer. The premise of this approach is that setting smaller neurons in the higher-level neural network will enable more abstract information to be learned from the data [20].

[1] https://ohollo.github.io/chord-extractor/

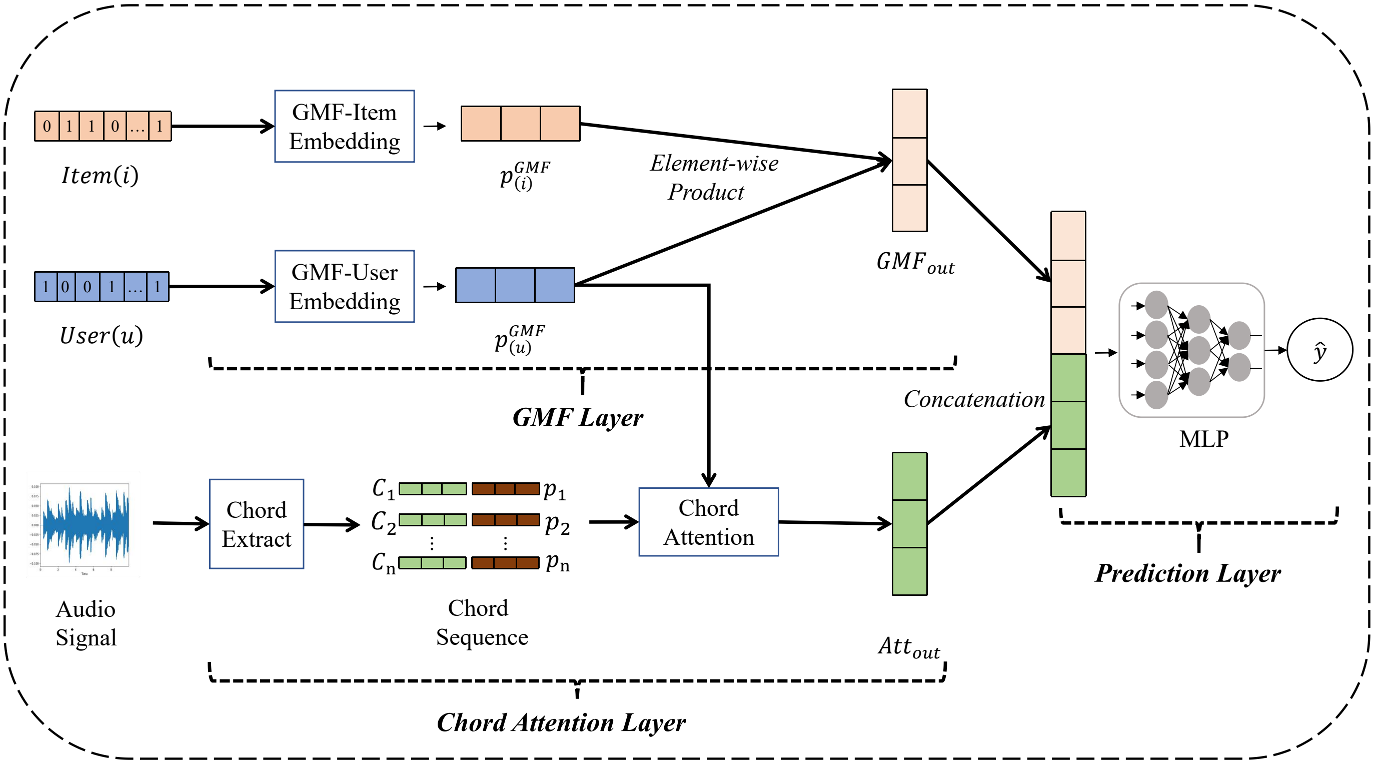

Fig. 1. The overall framework of our proposed COAT model

IV. Experiment Settings

In this section, we introduce the dataset used in the experiments. We then pose two research questions that we intend to answer in this paper to justify the proposed approach's effectiveness. Based on these questions, we design experiments and report their results.

A. Dataset Descriptions and Constructions

The experiments require a dataset containing the user's listening history and the song's audio file. For user listening history, we extracted a subset from the widely used last.fm-1b dataset [21]. As the dataset does not contain audio files of the songs, we downloaded the corresponding audio files from streaming platforms based on the collection of songs in the subset to form the dataset used in this paper.

Due to the size of the complete last.fm-1b dataset, conducting experiments on the complete data set would consume too much time. So we streamlined the last.fm-1b dataset in the following steps: Firstly, we used 2014 as the boundary to remove previous records from the complete dataset; Secondly, we filtered the top 10,000 popular songs from the song set to build a new subset; Finally, based on this subset, we removed listening records that were not relevant to it. We removed users with less than ten interaction records to ensure the dataset's quality. With 30,753 users, 10,000 songs and 1,533,245 interaction records, our dataset has a data sparsity of 99.50%.

B. Research Questions

- RQ1: Whether the chord attention mechanism improves the recommendation effect?
- RQ2: Whether the proposed method is better than traditional methods?

C. Experiment Design

To address the above research questions, we design four experiments accordingly.

- **Experiment 1:** To validate the effectiveness of our proposed attention mechanism, we conducted ablation experiments on it. As shown in Table I, we designed variants of the model with and without the attention mechanism (attention weights set to 1) and judged the effectiveness of the attention mechanism by comparing the performance of the recommendations.
- **Experiment 2:** We use a comparative experiment to verify whether our proposed model can obtain better results than the baseline approach. The chosen baseline methods include a traditional matrix factorization algorithm, a neural network-based collaborative filtering algorithm, and hybrid music recommendation algorithms based on audio features.

1) Parameter Settings.: The models involved in this paper use the same strategy for parameter settings. The range of searching for each hyperparameter is as follows: batch size is [128,256,512,1024], learning rate is [0.0001, 0.0005, 0.001, 0.005] and embedding size is [8,16,32,64].

2) Evaluation Protocols.: We used a strategy called leave-one-out to test the model's effectiveness, which has also been widely adopted in other work [22]. Regarding this strategy, the test set consists of one positive sample and several negative samples, where the positive sample is the last song in the user's listening record. For a given user, it would be time-consuming to add all songs in the dataset that have not been interacted with as negative samples to the test set for sorting, so we only sample 99 songs that have not been interacted with for a user as negative samples. This is a common strategy [23]. We use two common evaluation metrics to measure the effectiveness of ranking, *Hit Ratio* (HR) and *Normalized Discounted Cumulative Gain*(NDCG) [24]. The HR metric is given a 1 or 0 depending on whether the positive sample appears in the final top-n list. NDCG gives finer scores to positive samples based on where they appear in the top-n list. Higher scores are given to positive samples that appear higher up. We generate top-n recommendation lists for all users for each experiment round and use this to calculate two metrics, HR and NDCG. The average of all users' scores on both metrics is used as the final score of the model.

TABLE I
PERFORMANCE W.R.T. WITH OR WITHOUT ATTENTION

Embedding size	With Attention		W/o Attention	
	HR	NDCG	HR	NDCG
8	**0.581**	**0.344**	0.392	0.205
16	**0.629**	**0.390**	0.491	0.281
32	**0.647**	**0.421**	0.556	0.319
64	**0.653**	**0.442**	0.608	0.355

V. Results and Discussions

A. Whether the chord attention mechanism improves the recommendation effect (RQ1)

Table I shows the results of the ablation experiments related to the attention mechanism proposed in this paper. As can be seen from the table, the model's performance is always better when attention is applied than when it is not, and there is a significant drop in performance when attention is not applied. This phenomenon suggests that the application of attention mechanisms to distinguish the importance of different parts from multimedia content in deep learning-based recommendation algorithms can positively impact the effectiveness of recommendations.

B. Performance Comparison (RQ2)

In this experiment, we selected a traditional matrix factorization approach, a deep learning-based collaborative filtering approach, and a hybrid audio content approach as baselines for comparison with the COAT model.

- **WMF** [25]: The method uses a weighted matrix factorization technique to obtain embedding vector of users and items and produces recommendations from the inner vector product.

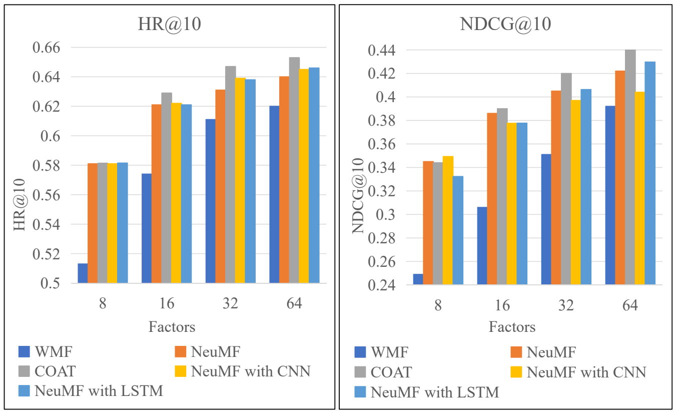

Fig. 2. Performance comparison of different methods.

- **NeuMF** [3]: The method uses deep neural networks to implement collaborative filtering and is the basis for many neural network-based recommendation algorithms.
- **NeuMF with CNN**: On top of NeuMF, CNN is used to process MFCCs features as a way to compare the performance of our proposed chord attention layer with that of the CNN-based approach.
- **NeuMF with LSTM**: On top of NeuMF, LSTM is used to process MFCCs features as a way to compare the performance of our proposed chord attention layer with that of the RNN-based approach.

Figure 2 shows the performance of each method under different predictive factors.

For the NeuMF method, the size of the predictive factor is the output dimension of the MLP and GMF layers in the method. For COAT, the size of the predictive factor is equal to the dimensions of the embedding vectors. For WMF, the number of predictive factors is equal to the embedding size. And for the LSTM-based approach and CNN-based approach, we use the Librosa [26] library to extract the MFCCs features from the audio to represent music content.

Figure 2 shows that the COAT model consistently achieves better results than the other methods on the two evaluation metrics when the predictor is greater than 8. NeuMF performs slightly better than the COAT model when the predictive factor size is 8. We consider this phenomenon because when the embedding dimension is too small, the COAT model has a smaller proportion of inputs related to user behavior, allowing the model to be influenced by the noise from the music content. This finding suggests that mining the user's behavioral history is crucial in designing recommendation algorithms. When the size of the neural network model is small, too much introduction may introduce more noise into the model and degrade the recommendation performance.

Compared with NeuMF, the LSTM-based approach can obtain some improvement, while the CNN-based approach will obtain more inferior results. This phenomenon indicates that audio features in matrix form, though similar to pictures, have more significant temporal features than local features. And because low-level audio features are complex, it is challenging to process them properly in music recommendation models.

As shown in Figure 2. The gap between the COAT model and other methods becomes larger as the predictive factor increases. In our analysis, the input chord sequence information makes the preference relationship between the user and the song more complicated, which means that the recommendation model needs to have stronger fitting power to learn this preference relationship. As the predictors increase, the model's width increases, making the model a stronger fitting capability. This phenomenon demonstrates again that when applying deep learning techniques to recommender systems, the use of larger-scale models brings improved recommendation results.

VI. Conclusions and Future Work

In this paper, we propose using higher-order music features instead of lower-order frequency domain features to represent music content in the music recommendation model and differentiate the importance of music content properly. To this end, we propose a music recommendation model known as COAT, which is based on chord progression and attention mechanism. The model integrates user song interactions and chord sequences of music and uses an attention mechanism to differentiate the importance of different parts of the song. The experimental results demonstrate the effectiveness of the method proposed in this paper.

In the future, we will explore introducing more higher-order music features, such as melody, rhythm, lyrics, etc., into the recommendation model based on the latest advancement in music information retrieval. At the same time, we will consider integrating this work with the existing graph neural network-based collaborative filtering recommendation model to achieve better recommendation results.

Acknowledgement

This work is partially supported by the Project of Beijing Municipal Education Commission (No.KM202110005025), the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions (CIT&TCD20190308), and Engineering Research Center of Intelligent Perception and Autonomous Control, Ministry of Education.

References

[1] M. Schedl, H. Zamani, C. Chen, Y. Deldjoo, and M. Elahi, "Current challenges and visions in music recommender systems research," *International Journal of Multimedia Information Retrieval*, vol. 7, no. 2, pp. 95–116, 2018.

[2] A. Van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Advances in neural information processing systems*, 2013, pp. 2643–2651.

[3] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T. Chua, "Neural collaborative filtering," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 173–182.

[4] J. Lee, K. Lee, J. Park, J. Park, and J. Nam, "Deep content-user embedding model for music recommendation," *arXiv: Information Retrieval*, 2018.

[5] M. Schedl, "Deep learning in music recommendation systems," *Frontiers in Applied Mathematics and Statistics*, vol. 5, 2019.

[6] P. Knees and M. Schedl, *Music similarity and retrieval: an introduction to audio-and web-based strategies*. Springer, 2016, vol. 9.

[7] W. Feng, T. Li, H. Yu, and Z. Yang, "A hybrid music recommendation algorithm based on attention mechanism," in *International Conference on Multimedia Modeling*. Springer, 2021, pp. 328–339.

[8] H.-T. Cheng, Y.-H. Yang, Y.-C. Lin, I.-B. Liao, and H. H. Chen, "Automatic chord recognition for music classification and retrieval," in *2008 IEEE International Conference on Multimedia and Expo*. IEEE, 2008, pp. 1505–1508.

[9] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T. 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*, 2017, pp. 335–344.

[10] Y. Yu, S. Luo, S. Liu, H. Qiao, Y. Liu, and L. Feng, "Deep attention based music genre classification," *Neurocomputing*, vol. 372, pp. 84–91, 2020.

[11] J. Yi, Y. Zhu, J. Xie, and Z. Chen, "Cross-modal variational auto-encoder for content-based micro-video background music recommendation," *IEEE Transactions on Multimedia*, 2021.

[12] D. Liang, M. Zhan, and D. P. Ellis, "Content-aware collaborative music recommendation using pre-trained neural networks," in *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015*. International Society for Music Information Retrieval, 2015, pp. 295–301.

[13] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys*, vol. 52, no. 1, p. 5, 2019.

[14] R. Wang, Z. Wu, J. Lou, and Y. Jiang, "Attention-based dynamic user modeling and deep collaborative filtering recommendation," *Expert Systems with Applications*, vol. 188, p. 116036, 2022.

[15] C. Zhou, J. Bai, J. Song, X. Liu, Z. Zhao, X. Chen, and J. Gao, "Atrank: An attention-based user behavior modeling framework for recommendation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018, pp. 4564–4571.

[16] Y. Du, T. Li, M. S. Pathan, H. K. Teklehaimanot, and Z. Yang, "An effective sarcasm detection approach based on sentimental context and individual expression habits," *Cognitive Computation*, pp. 1–13, 2021.

[17] Y. Liu, A. Pei, F. Wang, Y. Yang, X. Zhang, H. Wang, H. Dai, L. Qi, and R. Ma, "An attention-based category-aware gru model for the next poi recommendation," *International Journal of Intelligent Systems*, vol. 36, no. 7, pp. 3174–3189, 2021.

[18] J. Gong, S. Wang, J. Wang, W. Feng, H. Peng, J. Tang, and P. S. Yu, "Attentional graph convolutional networks for knowledge concept recommendation in moocs in a heterogeneous view," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 79–88.

[19] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based context for top- n recommendation with a neural co-attention model," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1531–1540.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[21] 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.

[22] I. Bayer, X. He, B. Kanagal, and S. Rendle, "A generic coordinate descent framework for learning from implicit feedback," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 1341–1350.

[23] A. Elkahky, Y. Song, and X. He, "A multi-view deep learning approach for cross domain user modeling in recommendation systems," in *Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 278–288.

[24] X. He, T. Chen, M. Kan, and X. Chen, "Trirank: Review-aware explainable recommendation by modeling aspects," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 2015, pp. 1661–1670.

[25] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Eighth IEEE International Conference on Data Mining*, 2009, pp. 263–272.

[26] B. Mcfee, C. Raffel, D. Liang, D. Ellis, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Python in Science Conference*, 2015.