Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

Edge-cloud computing oriented large-scale online music education mechanism driven by neural networks

Wen Xing[1*], Adam Slowik[2] and J. Dinesh Peter[3]

Abstract

With the advent of the big data era, edge cloud computing has developed rapidly. In this era of popular digital music, various technologies have brought great convenience to online music education. But vast databases of digital music prevent educators from making specific-purpose choices. Music recommendation will be a potential development direction for online music education. In this paper, we propose a deep learning model based on multi-source information fusion for music recommendation under the scenario of edge-cloud computing. First, we use the music latent factor vector obtained by the Weighted Matrix Factorization (WMF) algorithm as the ground truth. Second, we build a neural network model to fuse multiple sources of music information, including music spectrum extracted from extra music information to predict the latent spatial features of music. Finally, we predict the user's preference for music through the inner product of the user vector and the music vector for recommendation. Experimental results on public datasets and real music data collected by edge devices demonstrate the effectiveness of the proposed method in music recommendation.

Keywords Edge-cloud computing, Large-scale information fusion, Music recommendation, Neural networks

Introduction

With the rapid development of information technology, the education industry has undergone tremendous changes [1]. Online education has become an important way to improve the accessibility and effectiveness of education [2]. As one of the most popular forms of online education, online music education has become increasingly popular in recent years. However, the traditional centralized cloud computing model used in online music education faces challenges such as high latency, low security, and limited scalability. To address these challenges, an emerging paradigm called edge cloud computing has been proposed, which provides computing, storage, and networking resources at the network edge. The edge cloud computing model can significantly reduce latency, enhance security, and improve scalability.

With the development and progress of computer technology and artificial intelligence (AI), AI technology is gradually introduced in music education to promote online education. Initially, the application of AI in education led to the creation of an Intelligence Tutoring System (ITS) [3]. The primary focus of combining AI technology and education is the development of intelligent teaching systems, which represent the future of teaching and are the main subject of this paper. The rapid growth of information technology, along with the introduction and enhancement of new teaching system development models, has encouraged the comprehensive utilization of hypermedia technology, network infrastructure, and AI to create innovative teaching systems. ITS serves as

*Correspondence:
Wen Xing
xw@xxgc.edu.cn
[1] Xinxiang Institute of Engineering, Xinxiang 453700, China
[2] Department of Electronics and Computer Science, Koszalin University of Technology, Koszalin, Poland
[3] Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India

a prime example, incorporating domain, learner, and teacher models, encompassing the entire development of the teaching system. It boasts unparalleled advantages and significant appeal.

In the era of big data and cloud computing, the ability to sift through vast amounts of information and pinpoint students' interests poses a significant task and challenge for Internet education platforms [4]. Enter the recommendation system, which acts as a vital link between students and teachers and is increasingly integrated into major educational information service platforms. Its practical importance and theoretical research value cannot be overstated. Among the various multimedia product recommendation tasks, music recommendation has attracted extensive attention. However, the nature of music, encompassing emotions and styles, presents unique difficulties in quantification, description, and feature extraction, setting it apart from other internet products. Consequently, existing research in this domain still exhibits numerous shortcomings. Most studies focus only on audio, tags, or user feedback data, failing to fully leverage the wealth of available information.

Constructing a robust feature system solely based on a single category of data poses challenges, making the integration of multi-source information (e.g., edge and cloud data) crucial for music recommendation [5]. This integration holds immense significance in enhancing the recommendation process. Recommendation systems play a vital role in connecting students and teachers and are increasingly integrated into major educational services. However, music recommendation poses unique difficulties due to the abstract nature of music encompassing emotions and styles, making it difficult to quantify, describe and extract features compared to other products. Most existing research focuses solely on audio, tags or user feedback, failing to fully utilize the breadth of available data. Building a robust feature system using only a single data type is challenging, making the integration of multi-source information crucial for enhancing music recommendation. Therefore, there is a need for a large-scale online music education system leveraging deep learning and edge cloud computing to provide low-latency, high-bandwidth services while ensuring security and privacy [6, 7]. Advanced techniques like AI and machine learning can optimize the learning experience and improve the efficiency of music education. Specifically, a multi-source information fusion based model can address the cold start problem and improve recommendation effectiveness.

In this paper, we propose a large-scale online music education mechanism driven by the deep learning and edge cloud computing model. The proposed mechanism utilizes the edge computing infrastructure to provide low-latency and high-bandwidth services to music learners, while also ensuring the security and privacy of their data. The mechanism also leverages advanced technologies such as artificial intelligence and machine learning to optimize the learning experience and improve the efficiency of music education. Specifically, we propose a two-stage model based on multi-source information fusion for music recommendation, which solves the cold start problem at the music end and improves the recommendation effect. The contributions of this paper are listed as follows.

- We propose a two-stage model based on multi-source information fusion for music recommendation under the scenario of edge-cloud computing.
- We build a novel neural network model to fuse multiple sources of music information, including music spectrum extracted from audio, music label, artist ID, and release year information, to predict the latent spatial features of music.
- We conducted sufficient experiments on collected data and comparatively analyzed the performance of the models.

The rest of this paper is organized as follows. Section 2 reviews the related work on online music education and edge cloud computing. Section 3 introduces the proposed mechanism and its architecture. Section 4 presents the experimental results and performance evaluation of the mechanism. Finally, Sect. 5 concludes the paper and discusses future work.

Related work

The methods of the recommendation system mainly include traditional recommendation algorithms and improved traditional recommendation algorithms. In the development of artificial intelligence, recommendation algorithms combined with deep learning and reinforcement learning have emerged. In the long history of recommendation system development, traditional recommendation algorithms have played a key role in connecting the past and the future, and all emerging recommendation algorithms have been improved and innovated based on it. Today, under the background of artificial intelligence technology, the recommendation system has ushered in an era of continuous prosperity.

Researches on recommendation algorithms mainly includes traditional recommendation algorithms, recommendation algorithms improved by using technologies such as Singular Value Decomposition (SVD) and recommendation algorithms based on neural network that use deep learning for deep feature extraction. There is also a recommendation algorithm based on reinforcement

learning that uses reinforcement learning technology to effectively simulate the recommendation process to improve its accuracy and diversity.

In the study of traditional recommendation algorithms, Debnath et al. [8] proposed a hybrid recommendation system, and made the attribute weights based on content recommendation depend on their importance to users. The algorithm model balances the advantages and disadvantages of the two algorithms, and improves the recommendation effect of the recommendation system. Bagul et al. [9] proposed a content-based recommendation framework based on LDA and Shannon distance, which can generate recommendations similar to the documents provided at query time. He et al. [10] proposed a neural network-based collaborative filtering, which learns the interaction information of users and items through MLP, and uses MF to embed user information. Chen T et al. [11] proposed a context-based image recommendation method, which transfers the pixels of the image to the context for personalized image recommendation tasks. Selmene et al. [12] proposed a recommendation system based on user sentiment analysis, and integrated sentiment data into collaborative filtering, and used the average rating of items to replace the missing values of the rating matrix to improve recommendation accuracy. Sánchez et al. [13] integrated time dynamics and implicit feedback in the music recommendation system, and proposed a music recommendation method based on time perception. Fan et al. [14] proposed a recommendation method based on deep adversarial society, which uses a bidirectional mapping method and uses adversarial learning to transfer user information between the social domain and the commodity domain.

Research on music recommendation using traditional machine learning methods such as Gaussian mixture models, Bayesian networks, and hidden Markov models is common. Zheng et al. [15] proposed a dynamic music recommendation framework, which dynamically integrates the tag information and temporal of music tracks into user-item interaction to realize personalized music recommendation. Hu et al. [16] proposed the WMF algorithm for personalized TV program recommendation. Since then, this method has also been widely used in music recommendation. Liu et al. [17] used latent factor recommendation to recommend a list of background music songs for videos. Among them, the recommendation is based on the proposed scoring function, which contains the weighted average of video and music latent factors, and the objective function is designed by pairwise ranking method, and the objective function is optimized by stochastic gradient descent method. Li et al. [18] used the hidden Markov model to predict music sequences and recommend personalized music for users. Flexer et al. [19] studied the general problem of machine learning in high-dimensional spaces, that is, the impact of centrality on real-world music recommendation systems based on k-nearest neighbor graph visualization. They proposed mutual proximity graphs, which reduce the negative effect of centrality while improving accessibility. In addition, there are studies on converting audio data into bag-of-words representations for music recommendation [20].

In recent years, deep learning methods have gradually been adopted in the field of music recommendation. CNN is a deep feed-forward artificial neural network, including the LeNet-5 structure proposed by Lecun et al. [21], as well as the classic AlexNet [22] and VGG [23]. Van et al. [24] used convolutional neural networks for content-based music recommendation. Lee et al. [25] proposed a user-embedded music recommendation model, which combines user-music interaction information with music audio in an end-to-end manner to solve the music cold start problem. Recurrent Neural Network (RNN) can receive input of variable length and is good at processing sequence data. In recent years, many methods use LSTM [26], Gated Recurrent Unit (Gated Recurrent Unit, GRU) [27] and other RNN structures to better capture timing information and the process of user interest evolution. Balakrishnan et al. [28] used the LSTM model to extract the features of music audio and lyrics to calculate music similarity and achieve music recommendation.

Covington et al. [29] introduced DNN into YouTube recommendation, and achieved good results in realistic video recommendation. Bogdanov et al. [30] proposed a content-based recommendation method that can automatically generate users' music preferences directly from audio, and solve the visualization problem of users' music preferences by automatically inferring music avatars from semantic representations. Kiran et al. designed a hybrid recommendation system with deep learning, which uses embedding technology to integrate external information about users and items into a deep neural network, which alleviates the cold start problem [31]. Kim et al. [32] proposed a context-aware recommendation model, which combines convolutional neural network and probability matrix decomposition method, and can obtain the interrelated information before and after the document. Cantador et al. [33] clustered the semantics shared by users, established a multi-layer interest network according to the clustering results, and introduced it into collaborative filtering to improve the diversity of recommendations. Shani et al. [34] modeled the recommendation process as a Markov process, adjusted model parameters and continuously optimized recommendation results according to user feedback. Choi et al. [35] propose a reinforcement learning-based recommender system by using

dual clustering techniques with fewer states and actions, which can not only reduce space, but also effectively solve the cold-start problem, thereby improving recommendation quality.

There are also some music recommendation methods applied to specific scenarios. In order to cope with the real-time growth of data and realize online recommendation, Zou [36] proposed to use the Slope One incremental algorithm for music recommendation. Deng et al. [37] proposed a music recommendation method that integrates user emotions. Most works on music recommendation focus on the consumer side rather than the provider side, and Ren et al. [38] developed a bilateral approach for value-based music artist recommendation for streaming music scenarios. Chen et al. [39] improved the effect of music recommendation by studying the influence of social interaction on user preferences. The user's location often affects their music preference.

The proposed method

As mentioned above, in this paper we propose a two-stage model based on multi-source information fusion for music recommendation under the scenario of edge-cloud computing. The overall processing steps of the model are as follows. First, data is collected through edge collection devices and uploaded to the cloud server. Processing and transformation steps process the data into two parts, spectrogram and other formats. The model processes different types of data through two branch networks respectively. The first module mainly extracts features in the spectrogram through convolutional neural networks and recurrent neural networks. The second module extracts features through the encoding layer and the fully connected layer. Finally, the final output result is obtained through feature fusion. Specifically, we first use the music latent factor vector obtained by the Weighted Matrix Factorization (WMF) algorithm as the ground truth. Then, we build a neural network model to fuse multiple sources of music information, including music spectrum extracted from audio, music label, artist ID, and release year information, to predict the latent spatial features of music. Finally, we predict the user's preference for music through the inner product of the user vector and the music vector for recommendation. Figure 1 depicts the overall structure of the proposed model.

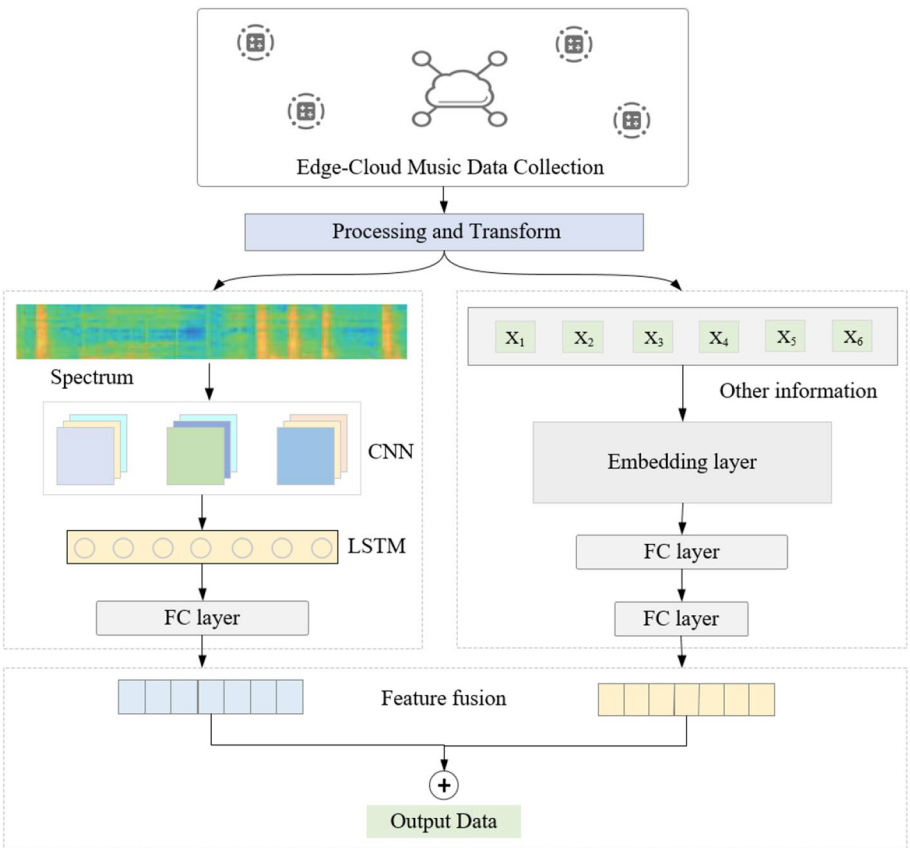

Fig. 1 The overall structure of the proposed model

Music feature extraction

The research motivation proposes developing a large-scale online music education system using an edge-cloud computing model. This refers to a distributed computing architecture combining edge computing and cloud computing. Edge computing provides low-latency and high-bandwidth services to end users by performing data processing at the edge of the network near the data source. For this music education system, edge servers closer to the learners can offer real-time interactive services and learning content while ensuring security and privacy of user data. Cloud computing provides flexible central storage and processing for large datasets. The cloud infrastructure allows aggregating and analyzing massive amounts of data from edge nodes to train machine learning recommendation models using deep learning algorithms. Together, the edge layer meets real-time interactive demands while the cloud enables robust analytics and personalization. This allows building an intelligent online education platform that can understand learners' interests and adapt offerings accordingly. In summary, the edge-cloud computing model provides a scalable architecture to support large numbers of online learners distributed across geographies while also harnessing large-scale data to optimize the learning experience.

Music is a kind of regular audio, and the music played or processed in the computer has different formats. The audio signal is formed by the superposition of sine waves of various frequencies and phases, which can be drawn into a continuously changing curve. The range of 20 Hz-20 kHz is the frequency of the audio signal that can be received by the human ear. Audio analysis is to process and analyze digital audio signals, and extract some special properties of audio signals in time and frequency. Different research fields are interested in different signal frequency ranges [40]. Before using a computer to analyze audio, it is necessary to perform audio sampling, that is, the process of discretizing continuous time. According to the Nyquist sampling theorem, when the sampling frequency is greater than 40 kHz, the sampled sound can achieve lossless sound quality. This sampling frequency is defined as twice the maximum sound frequency acceptable to humans, and the frequency to which humans are sensitive is about 4 kHz, so the sampling frequency is set at 8 kHz.

Multi-source information fusion refers to the combination of multiple information sources according to specific standards to obtain a consistent description or interpretation of the measured object, so that the information system has better performance than the subsystems it contains. We perform feature extraction and deep feature fusion on audio information, release time, label information, and singer identification information of music from the edge and cloud to obtain an effective prediction model. The overall framework of recommendation methods based on multi-source information fusion can be disassembled into two stages. In the first stage, we use the WMF algorithm to decompose the user-music listening times matrix to obtain the latent factor vectors of users and music. In the second stage, we use the neural network model to realize the multi-source information fusion of music and predict the latent factor vector of music. Finally, the inner product of the music vector and the user vector is used as the user's preference score for music and Top-N recommendations are made to solve the cold start problem of the music end.

In order to predict the user's preference for a certain music, an effective method is to map the user and the music to the same latent semantic space to realize the quantitative measurement of user preference. Therefore, in this paper, we use the weighted matrix factorization (WMF) algorithm proposed in the literature [16] to construct the latent semantic space, and obtain the latent factor vectors of users and music. The objective function of the weighted matrix factorization algorithm is shown in the following formula.

$$\min_{x+y_i} \sum_{u.i} c_{ui}(p_{ui} - \mathbf{x}_u^{\mathrm{T}}\mathbf{y}_i)^2 + \lambda\left(\sum_u \| \mathbf{x}_u \|^2 + \sum_i \| \mathbf{y}_i \|^2\right) \tag{1}$$

where p_{ui} is a binary preference variable, indicating user's preference for music i; $\mathbf{x}_u$ and $\mathbf{y}_i$ are latent factor vectors of user u and music i respectively; c_{ui} is a confidence variable, indicating the possibility of user u liking music i; $\lambda\left(\sum_u \| \mathbf{x}_u \|^2 + \sum_i \| \mathbf{y}_i \|^2\right)$ is the L2 regular term. The rating of music i for user u is represented by the product of their respective latent factor vectors $\mathbf{x}_u$ and $\mathbf{y}_i$. We implemented the above matrix factorization process using the implicit library and obtained latent factor vectors for users and music in the utilized dataset.

Learning based on neural network

Although the latent representation of users and music is obtained, and the user's preference for music is quantified, the cold start problem cannot be solved because the latent factor vectors obtained through WMF are all known music. To this end, in this paper, we construct a music feature system through multi-source information fusion, and carry out supervised learning of music features. However, the sequence processing ability of CNN is not as good as LSTM. Inspired by this, since audio is a sequence of data, the neural network we use in this paper combines the advantages of CNN and LSTM, and uses the Embedding layer to fuse other information of singers and music for music information mining

and personalized recommendation. Through the neural network to capture the latent features of music in multi-source information, we fuse the latent features of music learned in each multi-source information together to achieve multi-source information fusion. The audio feature extraction part abstracts the audio spectrogram into a 100-dimensional feature vector. In the Embedding part, the music label, singer ID and release time information are respectively embedded for training, and the embedded representation of the three information is obtained, and feature fusion and selection are performed through a multi-layer perceptron (MLP). Finally, the high-order features obtained from the two parts are concatenated, and the feature selection is performed through the fully connected layer to obtain the music latent factor vector. The following will elaborate on the audio feature extraction part, embedding part and output layer.

In the music feature extraction part, in order to reduce the complexity of audio data learning, we convert the audio data in wav format into Mel spectrum through short-time Fourier transform and Mel filter bank, and use Mel spectrum as the input of neural network model. First, the short-time Fourier transform is used for time-frequency conversion of audio, and the sound wave image is converted into a spectrogram. The short-time Fourier transform is as shown in the formula below.

$$STFT_Z(t,f) = \int_{-\infty}^{+\infty} [z(u)g(u-t)]\mathrm{e}^{-j2\pi fu}\mathrm{d}u \tag{2}$$

where z(t) is signal source and g(t) is the window function. The time window width used in audio framing is 1024 audio frames, the jump size is 512 audio frames, and the Hamming window is used to extract audio amplitude information. Finally, we reduce the dimensionality of the spectrogram through a set of Mel filters to obtain the Mel spectrogram, and use the logarithmic energy output by each filter as the input of the audio part of the deep learning model. The Mel filter bank sets several band pass filters $\mathrm{H}_m(\mathrm{k}), 1 \le \mathrm{m} \le \mathrm{M}$ in the spectrum range, where M is the number of filters. In the normal frequency, the resolution of the Mel filter bank in the low frequency part is relatively high, which is in line with the auditory characteristics of the human ear. We use 128 Mel filters to get a 128-dimensional Mel spectral vector.

In the audio feature extraction model, we continuously use three convolutional layers and LSTM layers for feature extraction, and finally use a fully connected layer for feature fusion and selection to obtain a 100-dimensional audio feature vector. The dimensions of the audio feature vectors are set to be the same as those of the latent factor vectors. Since the spectrogram has the properties of both audio sequences and graphs, we combine the advantages of both CNN and LSTM to provide users with personalized music recommendations. LSTM uses three gates, namely the update gate, the forget gate and the output gate, which are used to control the memory state of the LSTM unit.

In the embedding part, word embedding is often utilized in the field of Natural Language Processing (NLP) for vocabulary representation. The Embedding layer maps words to a low-dimensional space, reducing the training difficulty of the NLP model. Inspired by this idea, we use an embedding layer to incorporate music tags, release times, artist IDs, and other available identifying information. The Embedding layer can make the model more scalable, and adding the identification data obtained in the future to the embedding training can build a more complete feature system. Before embedding training, we need to represent the feature values in one-hot encoding. One-hot encoding uses an N-bit state register to encode N states. Let the feature domain of the model be $\mathrm{F} = [F_1; F_2; \cdots; F_n]$, and use one-hot encoding to represent the one-hot vector $\boldsymbol{o}_{F_i}^j$ corresponding to the j-th eigenvalue of feature F_i. In the embedding layer, each feature corresponds to an embedding matrix. For the feature F_i, the feature mapping relationship in the embedding layer is as follows.

$$e_{F_i}^j = \boldsymbol{o}_{F_i}^j \mathrm{W}_f \tag{3}$$

where $e_{F_i}^j$ is the embedding vector corresponding to the j-th eigenvalue in feature F_i; $\mathrm{W}_f \in \mathbb{R}^{F_i \times \mathrm{d}}$ is the weight matrix corresponding to feature F_i in the embedding layer, and d is the vector dimension set by the embedding layer. Finally, the embedding vector of the music is learned by a multi-layer perceptron and spliced together to obtain the input of the next layer.

The output layer of the neural network model splices the hidden representations obtained from the audio feature extraction part and the embedding part, and performs feature fusion and selection through a fully connected layer, and outputs the music latent factor vector predicted by the model. The neurons in the fully connected layer all use the ReLU activation function, as shown below.

$$\mathrm{ReLU}(x) = \max\{0, x\} \tag{4}$$

Suppose the output of the neuron a is O_a, and we can obtain the O_a through

$$O_a = \mathrm{ReLU}(\boldsymbol{W}_a^{\mathsf{T}} X_a + \boldsymbol{b}_a) \tag{5}$$

where X_a is the input vector of the neuron a, $\boldsymbol{W}_a^{\mathsf{T}}$ is the weights vector, and $\boldsymbol{b}_a$ is the bias vector. The loss function used in our model is mean square error, which is calculated by

$$L = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{y}_i')^2 \tag{6}$$

where n is the total number of samples, $\mathbf{y}_i$ is the real latent factor vector corresponding to music i, and $\mathbf{y}_i'$ is the latent factor vector of music i predicted by the model.

Experiments and results

In this section, we evaluate the proposed large-scale online music education mechanism driven by edge cloud computing model in this paper. First, we introduce data acquisition and processing and state the metrics used to evaluate the model. Second, we comparatively analyze the performance of the model and compare it with other state-of-the-art methods. In the experiment, each model is implemented using Pytorch, the optimizer uses Adam, the MLP layer uses two layers of fully connected layers, and the number of neurons is 512 and 100 respectively. We conduct experiments on a personal desktop computer. The operating system is Windows10×64-bit operating system, the processor is Intel(R) Core(TM) i9-9900 K CPU 3.60 GHz, and the number of processor cores is 16. The desktop computer comes with 32GB of RAM and an NVIDIA GeForce RTX 3070 graphics card.

Data process and metrics

In the experiment, the music data information we use comes from the Million Songs Dataset (MSD) [41], in which the user listening data comes from the Echo Nest Taste Profile Subset, a subset of MSD. About 30 s of audio samples are obtained from music websites in the metadata collection. The training data used in this paper are all from a subset of the same dataset. In addition, we collected 10,000 pieces of music data based on edge devices to obtain online users' music cleaning characteristics. First, we de-sparse the data, and extract 10,000 user listening records, and finally get a data set consisting of 10,000 users and 50,000 pieces of music. Due to the limitation of computing resources and time cost, we choose to extract a channel of random 3s wav format audio data during the experiment, and divide the audio data into a unified format.

In the weighted matrix decomposition stage, we use all the user listening records involved in 50,000 pieces of music to calculate the latent factor vectors of 50,000 pieces of music and 10,000 users. In the latent feature learning stage, we use 10,000 music pieces as the test set, and the remaining 40,000 pieces of music are used as the training set. When making Top-N recommendations, we use all users involved in the test set for calculation.

Similar to other literature studies, in this paper we mainly evaluate the model from two aspects, namely, the latent feature learning effect and the recommendation effect. The ability of a model to predict latent factor vectors is an important part affecting recommendation performance. Therefore, we use Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Squared Logarithmic Error (MSLE) evaluates the prediction effect of the neural network model. They are respectively defined as follows.

Table 1 The comparison results achieved by four different models

Method	MSE	RMSE	MAE	MSLE
AutoEncoder	1.247	1.104	0.528	0.142
CNN	1.201	1.096	0.513	0.135
CNN + LSTM	1.066	1.032	0.504	0.129
Embedding	1.415	1.190	0.586	0.161
Ours	0.917	0.958	0.482	0.117

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2 \tag{7}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2} \tag{8}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \mid y_i - y_i' \mid \tag{9}$$

$$\mathrm{MS}LE = \frac{1}{n}\sum_{i=1}^{n}(\log(y_i' + 1) - \log(y_i + 1))^2 \tag{10}$$

where y_i' is the predicted value of music i latent factor vector; y_i is the real value of music i latent factor vector. MSE and RMSE describe the error between the predicted value and the real value from the perspective of spatial distance. MAE is more robust to outliers. MSLE is more tolerant to the range of sample values. For the evaluation of recommendation effect, we use precision, recall and F1 value as evaluation indicators.

Performance comparison

When evaluating the effect of the model, we use the same training set and test set to train each model and calculate the evaluation index to ensure the authenticity and validity of the evaluation results. The prediction effect of latent factor vector of neural network model directly affects the learning of latent features by the model, so we use MSE, RMSE, MAE, and MSLE to comprehensively evaluate the effect of the model. Table 1 compares the results achieved by four different models.

It can be seen from Table 1 that the autoencoder model has the lowest performance, which shows that the model

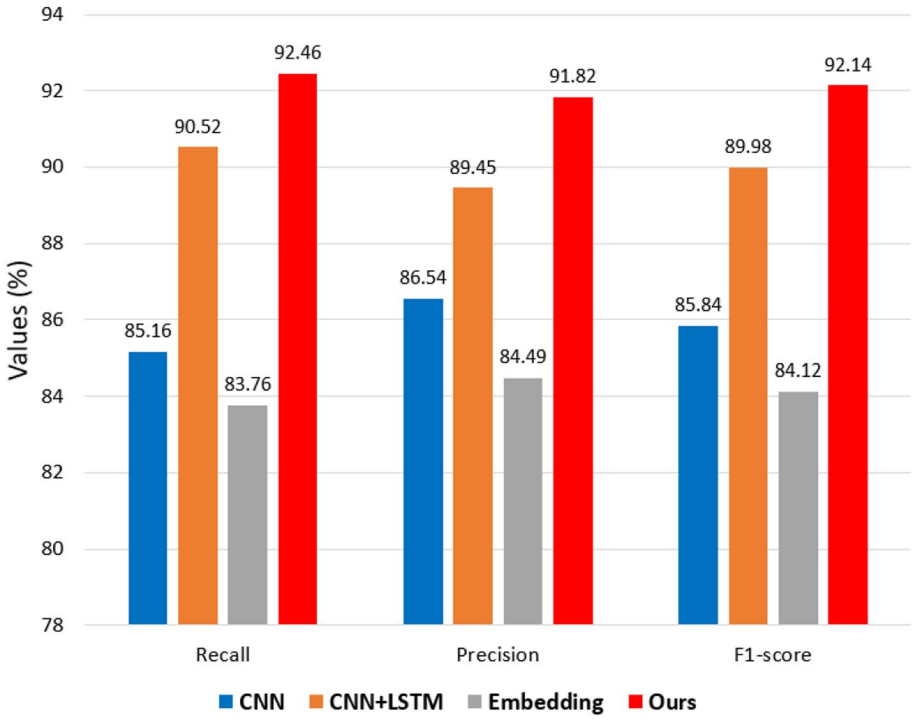

Fig. 2 The comparative results of the four models on the three indicators

is slightly less capable of extracting features in music data. After the CNN model is added to the LSTM structure, the prediction error decreases, which shows the effectiveness of the LSTM. This is due to the fact that the LSTM model is good at capturing temporal information, making up for the shortcomings of CNN. When using the embedding layer alone, the performance of the model is worse than that of the CNN model, and the four error indicators are all higher than CNN, which shows that the use of additional music information alone cannot surpass the performance of the neural network. Meanwhile, our proposed method achieves the best results. Since the audio information modeling is more complex, the effective information obtained from the audio feature learning module is insufficient, and it is necessary and effective to incorporate the release year, label and singer information.

Meanwhile, we validate the performance of the model on the recommendation task against these four models. Figure 2 depicts the comparative results of the four models on the three indicators. It can be seen from the experimental results that the accuracy, recall and F1 value of the method proposed in this paper are better than other models. The overall form shown in Fig. 2 basically matches that in Table 1, which also shows that the quality of feature extraction will also determine the classification performance of the model.

To investigate the impact of learning rate on model training and performance, we conducted experiments with four learning rates: 0.1, 0.01, and 0.001. Other hyperparameters were kept constant, along with the training and validation sets across experiments. Models were trained for 100 epochs while comparing the training loss, validation loss, and validation accuracy curves across learning rates. Figure 3 shows the model loss under different learning rates. With a learning rate of 0.1, the training loss decreased rapidly but overfitting occurred as evidenced by high validation loss. Setting the learning rate to 0.01 led to slower training loss reduction but lower validation loss. A learning rate of 0.001 resulted in steady training loss decrease and slightly improved validation performance over 0.01. However, a learning rate of 0.0001 caused slow training loss decline indicating underfitting. In summary, a learning rate of 0.001 achieved the optimal balance between training and validation loss, also yielding the highest validation accuracy. This learning rate ensures efficient model training while avoiding overfitting. Based on the comparative analysis, we set the learning rate to 0.001 for model training to guarantee both convergence speed and generalization capability. The experiment provides insights into selecting an appropriate learning rate schedule to optimize deep learning model performance.

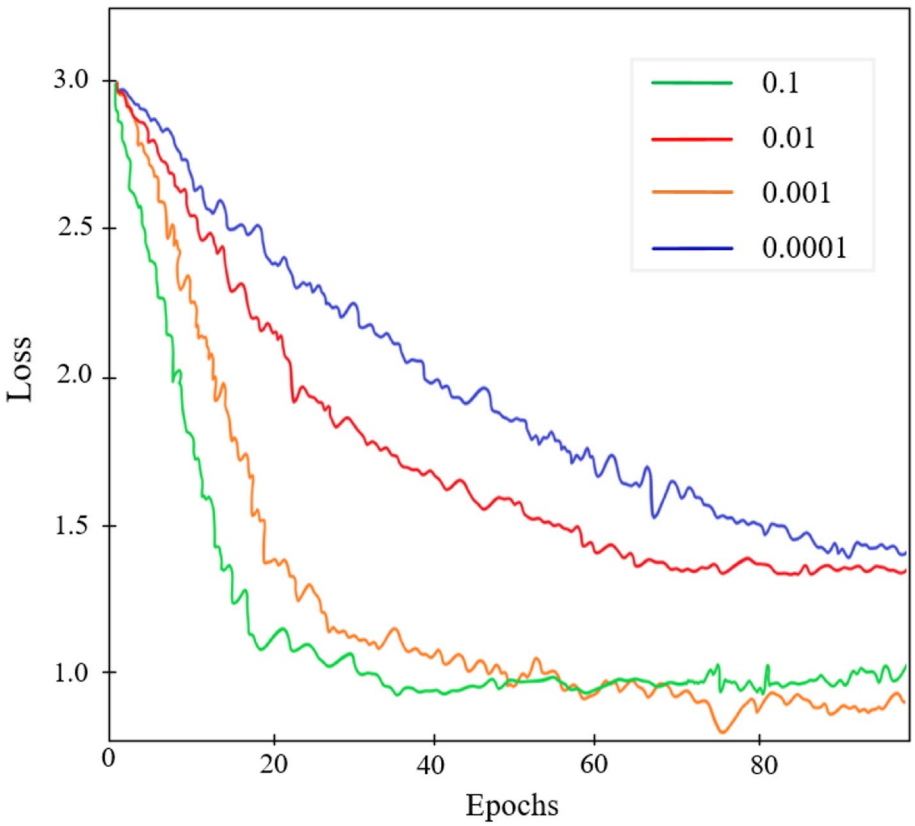

Fig. 3 The model loss under different learning rates

Conclusion

In this paper, we explore a large-scale online music education mechanism driven by an edge cloud computing model. Specifically, we construct a deep learning music recommendation method based on the idea of multi-source information fusion, which solves the cold start problem existing in traditional methods and provides a reference method for online music education. We first use the weighted matrix factorization algorithm to obtain the latent factor vectors of users and music to quantify users' preferences for music. Then we use CNN, LSTM and Embedding layers to realize the fusion of music audio and identification information, and predict the latent factor vector of music to solve the cold start problem of music. This paper conducts a large number of experiments on real music data sets, and the results show that the recommendation effect of the proposed method is better than other recommendation methods. The recommendation method constructed in this paper uses multi-stage recommendation and thus has greater limitations during training compared to end-to-end models. Therefore, in future work, the end-to-end music recommendation method should be further explored, and other audio information extraction methods should be explored to improve the personalized recommendation effect more substantially.

Authors' contributions
W.X. contributed to the writing and conception; A.S. contributed to the method; J.D.P. contributed to the data analysis and software.

Funding
This paper has not received any funding support yet.

Availability of data and materials
All data and materials will be available upon the reasonable request.

Declarations

Ethics approval and consent to participate
This declaration is "not applicable".

Competing interests
The authors declare no competing interests.

Received: 20 October 2023 Accepted: 25 November 2023
Published online: 07 March 2024

References

1. Yang Y (2020) Application of multimedia technology in vocal music digital teaching reform[C]//Journal of physics: Conference series. IOP Publishing 1648(4):042005

2. Castro MDB, Tumibay GM (2021) A literature review: efficacy of online learning courses for higher education institution using meta-analysis. Educ Inform Technol 26:1367–1385
3. Anderson JR, Boyle CF, Reiser BJ (1985) Intelligent tutoring systems. Science 228(4698):456–462
4. Khan M, Naz S, Khan Y et al. (2023) Utilizing machine learning models to Predict Student Performance from LMS Activity Logs. IEEE Access
5. Cai H, Xu B, Jiang L et al (2016) IoT-based big data storage systems in cloud computing: perspectives and challenges. IEEE Internet Things J 4(1):75–87
6. Rafique W, Shah B, Hakak S et al (2023) Blockchain Based Secure Interoperable Framework for the Internet of Medical Things[C]//Proceedings of International Conference on Information Technology and Applications: ICITA 2022. Singapore: Springer Nature Singapore, : 533–545
7. Rafique W, Khan M, Khan S, Ally JS (2023) SecureMed: A Blockchain-Based Privacy-Preserving Framework for Internet of Medical Things. Wireless Communications and Mobile Computing 2023:2558469. https://doi.org/10.1155/2023/2558469
8. Debnath S, Ganguly N, Mitra P (2008) Feature weighting in content based recommendation system using social network analysis[C]//Proceedings of the 17th international conference on World Wide Web. : 1041–1042
9. Bagul DV, Barve S (2021) A novel content-based recommendation approach based on LDA topic modeling for literature recommendation[C]//2021 6th International conference on inventive computation technologies (ICICT). IEEE, : 954–961
10. He X, Liao L, Zhang H et al. (2017) Neural collaborative filtering[C]//Proceedings of the 26th international conference on world wide web. : 173–182
11. Chen T, He X, Kan MY (2016) Context-aware image tweet modelling and recommendation. Proceedings of the 24th ACM international conference on Multimedia, p 1018–1027. https://doi.org/10.1145/2964284.2964291
12. Selmene S, Kodia Z (2020) Recommender System Based on User's Tweets Sentiment Analysis. Proceedings of the 4th International Conference on E-Commerce, E-Business and E-Government, p 96–102. https://doi.org/10.1145/3409929.3414744
13. Sánchez-Moreno D, Zheng Y, Moreno-García MN (2018) Incorporating time dynamics and implicit feedback into music recommender systems. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, Chile, p 580–585. https://doi.org/10.1109/WI.2018.00-34
14. Fan W, Derr T, Ma Y et al (2019) Deep adversarial social recommendation. arXiv preprint arXiv:1905.13160,
15. Zheng E, Kondo GY, Zilora S et al (2018) Tag-aware dynamic music recommendation. Expert Syst Appl 106:244–251
16. Hu Y, Koren Y, Volinsky C (2008) Collaborative filtering for implicit feedback datasets[C]//2008 Eighth IEEE international conference on data mining. IEEE 15:263–272
17. Liu CL, Chen YC (2018) Background music recommendation based on latent factors and moods. Knowl Based Syst 159:158–170
18. Li T, Choi M, Fu K et al. (2019) Music sequence prediction with mixture hidden markov models[C]//2019 IEEE International Conference on Big Data (Big Data). IEEE, : 6128–6132
19. Flexer A, Stevens J (2018) Mutual proximity graphs for improved reachability in music recommendation. J new Music Res 47(1):17–28
20. McFee B, Barrington L, Lanckriet G (2012) Learning content similarity for music recommendation. IEEE Trans Audio Speech Lang Process 20(8):2207–2218
21. LeCun Y, Bottou L, Bengio Y et al (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
22. Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. Commun ACM 60(6):84–90
23. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556,
24. Van den Oord A, Dieleman S, Schrauwen B (2013) Deep content-based music recommendation. Adv Neural Inf Process Syst 26:104811
25. Lee J, Lee K, Park J et al (2018) Deep content-user embedding model for music recommendation. arXiv preprint arXiv:1807.06786,
26. Graves A (2012) Supervised Sequence Labelling with Recurrent Neural Networks. Studies in Computational Intelligence (SCI, Volume 385), Springer. https://doi.org/10.1007/978-3-642-24797-2
27. Chung J, Gulcehre C, Cho KH et al (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412:3555,
28. Balakrishnan A, Dixit K (2014) Deepplaylist: using recurrent neural networks to predict song similarity. Stanford University, p 1–7. https://cs224d.stanford.edu/reports/BalakrishnanDixit.pdf
29. Covington P, Adams J, Sargin E (2016) Deep neural networks for youtube recommendations[C]//Proceedings of the 10th ACM conference on recommender systems. : 191–198
30. Bogdanov D, Haro M, Fuhrmann F et al (2013) Semantic audio content-based music recommendation and visualization based on user preference examples. Inf Process Manag 49(1):13–33
31. Kiran R, Kumar P, Bhasker B (2020) DNNRec: a novel deep learning based hybrid recommender system. Expert Syst Appl 144:113054
32. Kim D, Park C, Oh J, et al (2016) Convolutional matrix factorization for document context-aware recommendation[C]//Proceedings of the 10th ACM conference on recommender systems 233–240
33. Cantador I, Castells P (2011) Extracting multilayered communities of interest from semantic user profiles: application to group modeling and hybrid recommendations. Comput Hum Behav 27(4):1321–1336
34. Shani G, Heckerman D, Brafman RI et al (2005) An MDP-Based Recommender System. J Mach Learn Res 6(43):1265–1295
35. Choi S, Ha H, Hwang U et al (2018) Reinforcement learning based recommender system using biclustering technique. arXiv preprint arXiv:1801.05532,
36. Zou W (2018) Design and application of incremental music recommendation system based on slope one algorithm. Wireless Pers Commun 102:2785–2795
37. Deng S, Wang D, Li X et al (2015) Exploring user emotion in microblogs for music recommendation. Expert Syst Appl 42(23):9284–9293
38. Ren J, Kauffman R, King D (2019) Two-sided value-based music artist recommendation in streaming music services
39. Chen J, Ying P, Zou M (2019) Improving music recommendation by incorporating social influence. Multimedia Tools and Applications 78:2667–2687
40. Chen CH, Sühn T, Kalmar M et al (2019) Texture differentiation using audio signal analysis with robotic interventional instruments[J]. Comput Biol Med 112:103370
41. Bertin-Mahieux T, Ellis DPW, Whitman B et al (2011) The million song dataset

Publisher's Note