International Journal of Ambient Computing and Intelligence
Volume 14 • Issue 1

BTSAMA:
A Personalized Music Recommendation Method Combining TextCNN and Attention

Shaomin Lv, College of Music and Dance, Yulin Normal University, China*
Li Pan, UCSI University, Malaysia

ABSTRACT

To deal with the problems of occurring personalized music recommendation methods, for instance, low explanation, low accuracy of recommendation, and difficulty extracting information effectively, a personalized music recommendation method combining TextCNN and attention is proposed. Firstly, TextCNN model and BERT are combined to capture local music continuous features. Secondly, self-attention is introduced to solve the remaining omitted non-continuous features that are not paid attention by TextCNN. Finally, multi-headed attention mechanism is used to get features of hotspot music and user's interest music, and cascading fusion method is used to achieve click prediction. Experimentally, the proposed model can effectively recommend personalized music, its MAE values on FMA and GTZAN datasets are 0.156 and 0.146, respectively, improving by at least 6.6% and 3.3% compared to other comparative models. And its RMSE result values on the FMA and GTZAN datasets are 0.185 and 0.164, respectively, improving by at least 12.4% and 5.2% compared to other comparative models.

KEYWORDS

BERT model, Multiple attention mechanism, Personalized music recommendation, Self attention mechanism, TextCNN

BTSAMA-A PERSONALIZED MUSIC RECOMMENDATION METHOD COMBINING TEXTCNN AND ATTENTION

The Internet has developed rapidly in the 21st century, and the exponential expansion of data volume has become a trend (Garcia-Gathright et al., 2018; Gunawan & Suhartono, 2019). In line with the 47th Statistical Report on the Development of China's Internet, it is known that there were about one billion Internet users in China by the end of 2020, and the Internet penetration rate is as high as 71% (Wang et al., 2018; Wei et al., 2019; Zhao et al., 2020). Consequently, the digital music market is huge, and it is known from the 2020 China Music Industry Development Report that the growth rate

DOI: 10.4018/IJACI.327351 *Corresponding Author

of the online music industry reaches more than 8%. After years of development and improvement (Liu et al., 2019; McInerney et al., 2018; Melchiorre et al., 2021), the recommendation technology is used widely in various fields. Most people search for some previously known artists or preferred song categories by using the search function of the software. Accurate personalized recommendations for the user's favorite songs can better enhance the user's stickiness to the platform (Drott 2018; Kim & Kim, 2018).

Music has a common and obvious phenomenon, which is information overload. On the one hand, for users, if they do not know what types of songs they like, it is almost impossible for them to sample every song to record their favorite songs in the face of massive music libraries; so how to pick out their favorite songs from the mass of songs is time-consuming and labor-intensive (Abdul et al., 2018; Bauer & Schedl, 2019; Millecamp et al., 2019). Even if users provide clear preferences, how should music service platforms go about generating user preferences? This is also something that needs to be researched and explored. Data engineers at Spotify (Li & Zhang, 2018; Wang et al., 2020; Zheng et al., 2018), a well-known music service platform, used data analysis tools based on backend log files to find that 80% of users listen to the same 20% of songs, and the remaining 80% of songs are hardly ever played, a long-tail effect (Kowald et al., 2020; Millecamp et al., 2018) that has been seen time and again in other fields. On the other hand, for music service platforms, with today's trend of cultural diversity, user preferences vary widely toward differentiation (Kim et al., 2019; Prey, 2018; Sachdeva et al., 2018). How the major music service platforms can easily and accurately retrieve music that meets users' individual needs from the huge music library and reduce their search time and audition time is a difficult hurdle for music platform providers to overcome. In addition, music platforms providing differentiated recommendation results are also significant in promoting the spread of cold songs and increasing the variety of users' favorite songs, thus increasing user stickiness (Ayata et al., 2018; Jin et al., 2018; Kouki et al., 2019). Therefore, a personalized music recommendation system was born in such a contradictory context. For the purpose of meeting the personalized needs of different user groups for music, major well-known music platforms have launched their own personalized music recommendation systems, which have been noticed by users and have increased certain user stickiness (Chen et al., 2018; Karakayali et al., 2018; Werner, 2020; Zhao et al., 2019).

For example, Last.fm (Liebman et al., 2019) and Pandora (Jin et al., 2019), which were created in foreign countries, have attracted a host of users and arguably have the largest user base in the field of personalized music recommendation. The domestic NetEase Cloud Music and Douban Radio have also grown rapidly and are liked by the public. However, users are seeking more personalized and diverse recommendation results, which forces researchers to explore information beyond users and songs (i.e., contextual information). A system that can provide users with personalized song recommendations is important as it analyzes data related to contextual information; this further explores the various scenarios in which users listen to songs and analyzes the indirect effects of each scenario on users' preferences. It also adopts appropriate ways to integrate contextual information into the recommendation system to offer users with more accurate and diverse personalized recommendations. Therefore, a personalized music recommendation method combining TextCNN and attention is proposed; experimentally, the proposed method significantly outperforms several other advanced approaches in personalized music recommendation.

RELATED WORK

At present, there are three main personalized recommendation algorithms: the collaborative filtering-based recommendation algorithm, the content-based recommendation algorithm, and the singular value decomposition (SVD)-based recommendation algorithm. Although the content-based personalized recommendation algorithm has stable recommendation results, it is constrained by the information retrieval technology, has limited feature extraction ability, has a narrow recommendation range, and

cannot explore the potential learning information of users; moreover, the collaborative filtering-based personalized recommendation is the most widely used personalized recommendation algorithm at present, and it has better recommendation performance, but the algorithm has poor recommendation effect when encountering problems – for instance, cold start and sparse data. The SVD-based personalized recommendation algorithm is essentially a matrix decomposition operation; at the same time it has two fatal shortcomings: completing the sparse scoring matrix and high computational complexity of high-dimensional matrix decomposition. Deep learning techniques provide new ideas and solutions for the further improvement and development of personalized music recommendation systems. Literature (Li et al., 2007) proposed to group music effectively and use probabilistic models to process user ratings for recommendation by collaborative filtering, but the recommendation accuracy is low. A study by Zhang et al. (2019) proposes a recommendation method with deep variational matrix decomposition for large-scale sparse data sets and obtains the latent features of users and items through deep nonlinear structures. A hybrid recommendation model that is based on multi-objective optimization is proposed in a study by Cai et al. (2020). A study by Zhang et al. (2019) proposed a deep collaborative filtering recommendation model that captures high-dimensional nonlinear features by MLP before embedding them in the output layer using a similarity adaptive corrector to correct the accuracy of prediction. A study by Wei et al. (2017) proposes a collaborative filtering approach based on deep learning and temporal awareness by closely coupling collaborative filtering and DNN. A study by Guo et al. (2017) proposes a neural network model Deep FM that incorporates deep learning and FM techniques for recommendation systems and advertising click-through rate prediction. A study by Zhao et al. (2018) proposes a deep reinforcement learning-based page recommendation framework to recommend page items to users and provide feedback by efficiently acquiring dynamic information about recommendation interactions. A study by Huang et al. (2021) simulates the mutual effect on the recommender system and the user by RNN and uses reinforcement learning to optimize the recommendation model but with low efficiency. A study by Chen et al. (2019) imitates the dynamic behavior of users and learns their rewards for recommendation by generating adversarial networks, and a cascaded DQN is proposed to obtain a recommendation strategy that can efficiently handle a large number of combinations of candidate sets. A study by Mao et al. (2022) proposes a Music CRN for personalized music recommendation by learning the audio content characteristics of music to facilitate music classification and recommendation and converts audio into sound spectrum graph "images" by Fourier transform. A study by Weng et al. (2022) proposed a graph-based attentional sequential model (GASM) for metadata. A personalized music recommendation algorithm based on IR-MC was proposed by Tofani et al. (2022), but the recommendation accuracy was not high.

The above methods have low recommendation accuracy and low interpretability: firstly, for the audience, the working mechanism of some music recommendation models is not transparent, which brings out insufficient persuasiveness of recommendation results. Secondly, many existing methods only consider user history click music without considering popular music, and recommendation performance is easily affected by data scarcity and user cold start issues. Thirdly, many existing deep learning-based personalized music recommendation models are difficult to balance local music continuous and non-continuous features, resulting in weak recommendation, a large loss of audience, and the low loyalty of users. To overcome the above problems, a personalized music recommendation method combining TextCNN and attention is proposed:

1. Few existing deep learning-based music recommendation methods consider hot music, while the proposed model considers both user history clicked music and hot music, which can better ease the data scarcity and cold start problems
2. Combining the BERT model and TextCNN model to classify music reviews, speciating the review text and capturing local music continuous features, the effective combination of both can acquire characteristic information that contributes more to music recommendation.

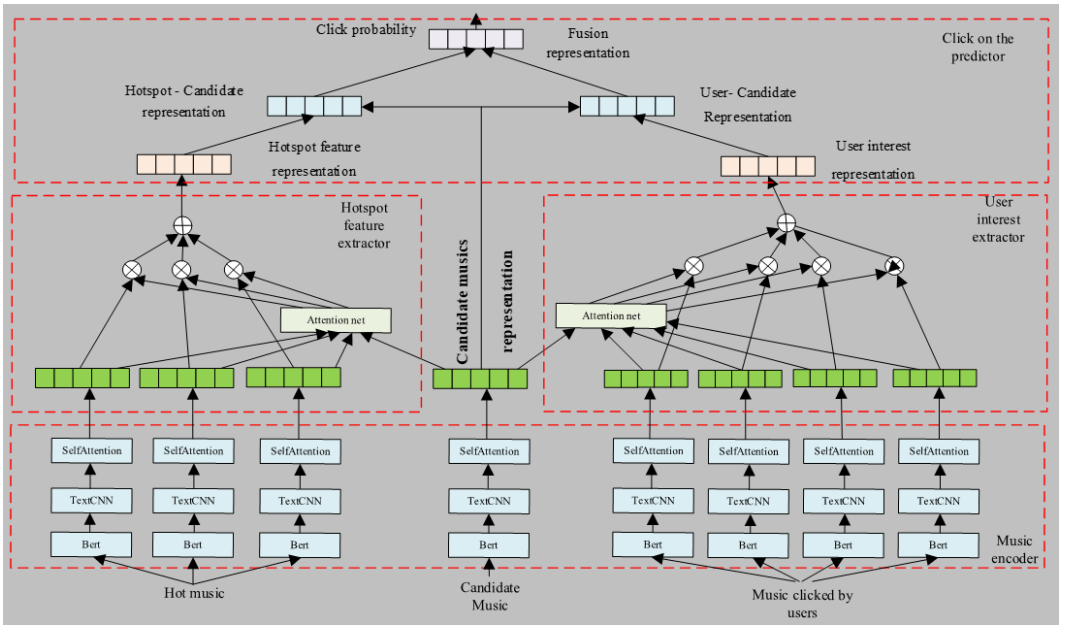
3. Self-attention is introduced to address the remaining omitted non-contiguous features that are not attended to by TextCNN, and the fused sequences are combined to obtain the user's interest features through a multi-headed attention mechanism network to calculate its preference value with each candidate music; this effectively enhances the personalized music recommendation performance of the model.

PROPOSED PERSONALIZED MUSIC RECOMMENDATION ALGORITHM

BTSAMA Personalized Music Recommendation Model

The Heterogeneous Graph Attention Network (HAN) contains four main components: a music encoder, a hot feature extractor, a user interest extractor, and a click predictor. The input of HAN includes some hot music in the past time, the click history of a certain user, and a candidate music. For each music, its music representation can be obtained by the music encoder. To improve the extraction ability of text features, this model introduces BERT to convert word sequences into vector sequences for pre-training of word embedding, and it applies a self-attentive network to extract discontinuous text features to make up for the limitation that TextCNN can only extract local continuous features. On the basis of obtaining each music representation, in order to extract hot features based on hot music representations, a hot feature extractor is designed, which uses a Multi-Head Attention Network (MHAN) to focus on the hot music related to the candidate music, thus enabling dynamic aggregation of hot music representations using different weights. To extract user interest features, a user interest extractor is used to aggregate user click histories. Based on the obtained hotspot feature representation and user interest representation, in order to predict the click probability of users for candidate music, a click predictor is proposed, which can fuse hotspot feature representation, user interest representation, and candidate music representation to balance user interest and hotspot features and finally predict the click probability of candidate music. The general structure diagram of the proposed method is illustrated in Figure 1.

Figure 1. Overall structure of the proposed BTSAMA



BERT Model

The BERT model uses a bidirectional transformer encoder to generate the language model. This bi-directional feature allows the model to learn based on both sides of the word. The input vector of BERT is generally set to 512 length and consists of three embedding features:

1. Marker embedding: Each word is assigned a 768-dimensional vector through a query.
2. Segment embedding: The segment embedding layer is used to assist the BERT model in classifying input sentence pairs where all words in the first sentence are assigned a value of 0 and the second sentence is assigned a value of 1.
3. Position embedding: In the process of word vector encoding, position encoding features are introduced and their information is concatenated with the word vector to solve the problem of non-capturing sequence position information for transformer.

The model is suggested to fine tune, which leads the pre-trained BERT model is applied into music recommendation tasks. With the powerful encoding ability of transformer, the embedding word vectors are generated, which will be input into TextCNN.

TextCNN Model

Next, the embedding word vectors are input into TextCNN; with its powerful feature extraction ability, TextCNN is utilized to capture local music continuous features. Figure 3 exhibits the specific vector operation process. For the text domain, local features can be regarded as sliding windows (N-gram) consisting of multiple words, and convolutional neural networks can effectively avoid the problems of exponential growth of parameter space with increasing n and data scarcity that may be brought by traditional N-gram methods. Complex smoothing mechanisms are required.

The specific processing idea is shown in Figure 3, where the fused vectors (referred to as token in the following), after being processed by the dynamic word vector representation module (i.e., two-layer transformer block), are vertically stitched by position as the input to the multi-channel convolutional neural network module. A sentence consisting of N tokens can be denoted as $x_{1:n} = x_1 \oplus x_2 \oplus ...x_n$, where $\oplus$ represents vertical splicing. The input is an N × k matrix, and k represents the dimension size of the word embedding, which is computed using the one-dimensional convolutional processing. First, a filter $W(W \in R^{h\times k})$ is used to slide from top to bottom on the input matrix, and when it reaches a certain position, the array inside the filter W is convolved with the array inside the window $x_{i:i+h-1}$ of the input matrix covered (i.e., the result obtained by multiplying element by element is then summed to get a feature $c_i = f(w^* x_{i:i+h-1} + b)$, where $x_{i:i+h-1}$ represents a window of size h × k consisting of the i_{th} row to the i_{th} + h-1_{th} row of the input matrix, specifically stitched by $x_i, x_{i+1},..., x_{i:i+h-1}$, *h* indicates the number of words in the window; the h × k-dimensional weight matrix is denoted as W, the bias parameter is denoted as b, and the nonlinear function is denoted as f.

Self-Attention Mechanism

Because TextCNN focuses on the local continuous features, some distant key information in music lyrics text may be omitted. For example, in the lyrics 'Listening to this song, I can see a picture in my mind: a group of nomads came to a desolate place, where there was a war not long ago and there were many corpses everywhere. The sad atmosphere was filled with sadness', 'desolate' and 'war' are keywords with important information, with a distance of five words between them; the ability of TextCNN to capture features with distance discontinuity is not significant (discontinuous features refer to extracting features from several non-adjacent words, and the distance between keywords is generally greater than the size of the convolutional kernel). A study by Lin et al. (2017) proposed to

Figure 2. BERT pre-training model

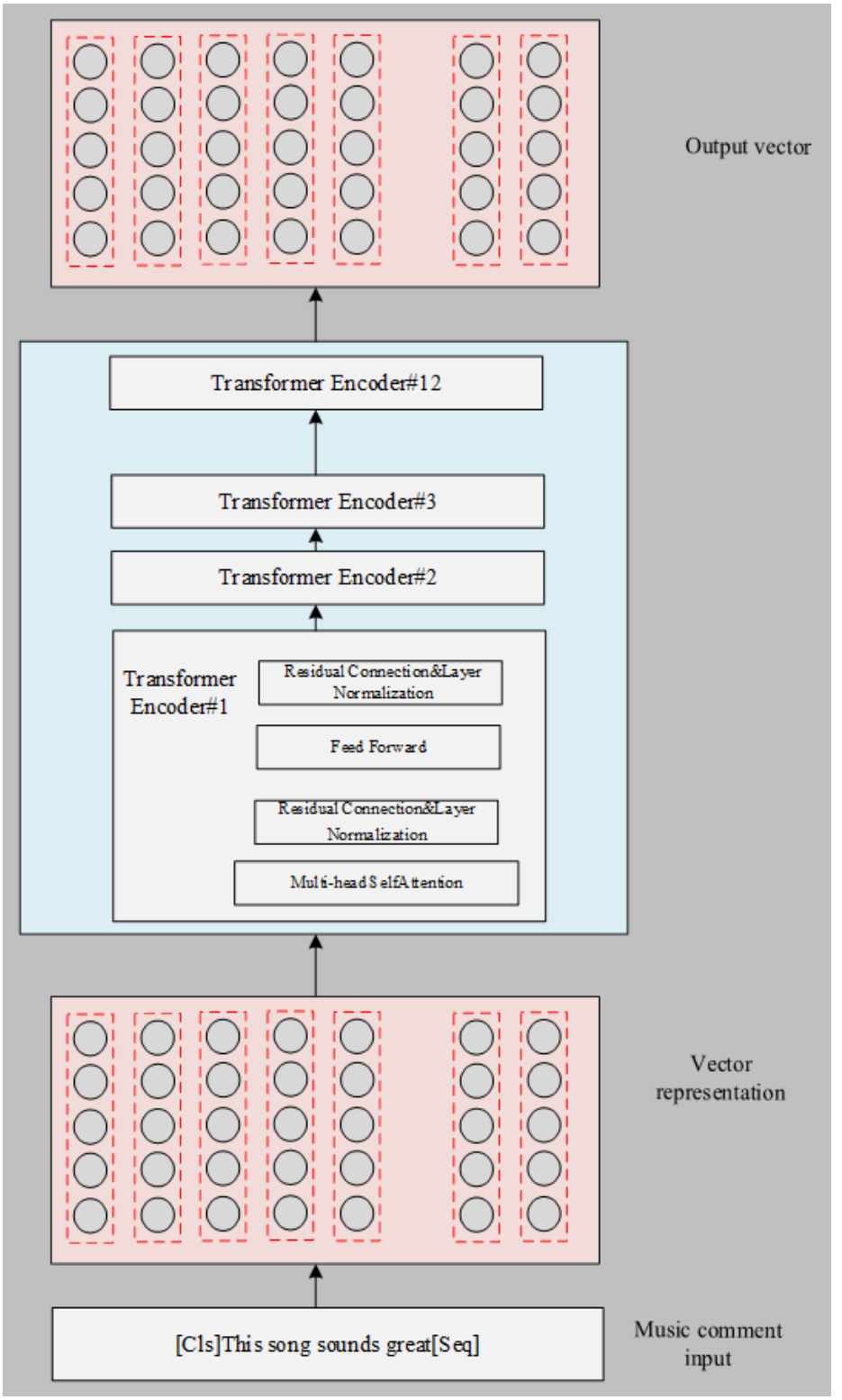
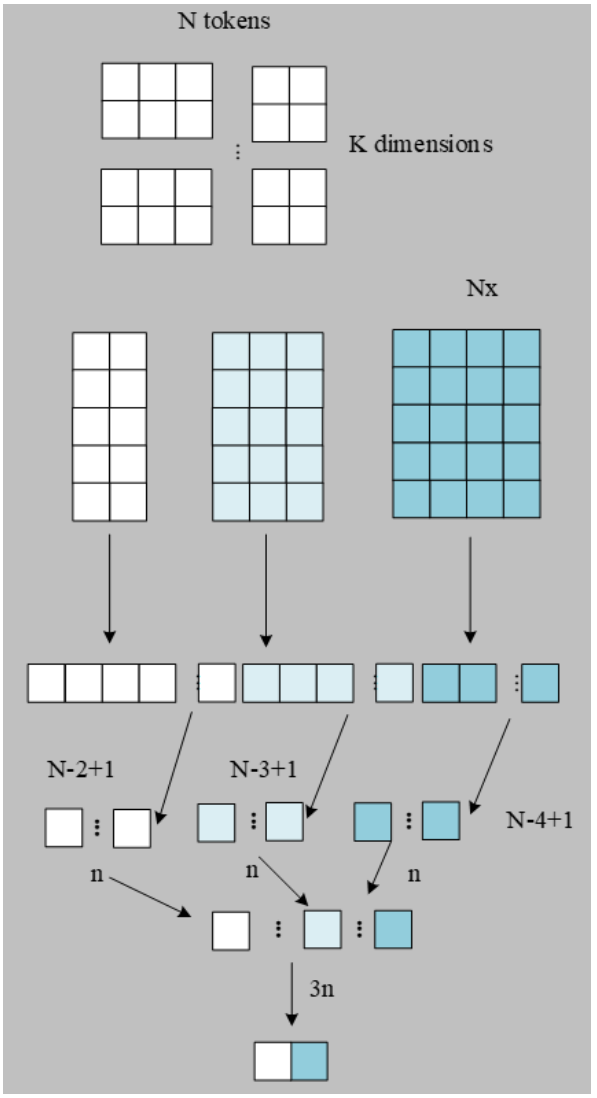


Figure 3. TextCNN classification model



introduce the self-attention network to capture the discontinuous features of sentences using BiLSTM, and it accomplishes good results. Inspired by this work, the remaining omitted non-contiguous features that are not attended by TextCNN are addressed by introducing self-attention. The self-attention mechanism is a special attention mechanism that better captures the relevance within the data features relative to the initial attention mechanism and reduces the dependence on the need for other external information. It enables the neural network model to be intent upon the important parts of the input data or features flexibly and automatically according to its own needs, which enhances the expressiveness of the model greatly. In order to simplify, and avoiding loss of generality, the authors omit the layer identification, and, as in the transformer encoder part, the self-attention mechanism contains three feature matrices Q, K, and V, and the three matrices are identical and denoted by $\boldsymbol{H}$.

$$Self - Attention(\boldsymbol{H}) = \tilde{\boldsymbol{A}}\boldsymbol{H}\boldsymbol{W}^{v} \tag{1}$$

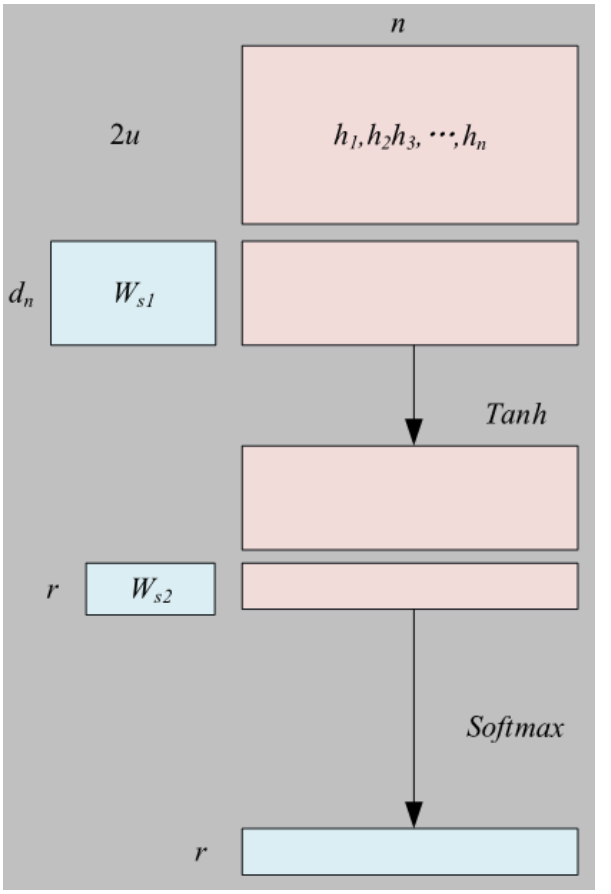
$$\tilde{\boldsymbol{A}} = soft\max(\frac{\boldsymbol{HW}^{\boldsymbol{Q}}(\boldsymbol{HW}^{\boldsymbol{K}})^{T}}{\sqrt{d_k}}) \tag{2}$$

where $\boldsymbol{W}^{v}$, $\boldsymbol{W}^{\boldsymbol{K}}$, and $\boldsymbol{W}^{\boldsymbol{Q}}$ are the parameter matrices; $soft\max$ is the normalization of the matrix by rows.

Multi-Head Attention Mechanism

To be better intent upon the hot music related to the candidate music, a multi-headed attention network is introduced. The multi-headed self-attentive mechanism stacks multiple self-attentive modules in parallel rather than sequentially backward and forward and aggregates them later. Different parallel self-attentive modules with the same input data help capture feature information at different levels in the input data and improve the representational power of the model. Similar to multiple filters in CNN to learn different expressive information of pictures, as shown in Figure 5, Q, K, and V are first computed by Scaled Dot-Product Attention after different h linear projections, which can learn different semantic information. The equation (3) represents the conversion of the results of multiple self-attentive heads into a dimension-specific output vector after stitching, and the computation process of each multi-head module is represented by equation (4).

Figure 4. Module diagram of self-attention mechanism



$$MultiHead(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = Concat(head_i,...,head_h)\boldsymbol{W}^O \tag{3}$$

$$head_i = Attention(\boldsymbol{Q}\boldsymbol{W}_i^Q,\boldsymbol{K}\boldsymbol{W}_i^K,\boldsymbol{V}\boldsymbol{W}_i^V) \tag{4}$$

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = soft\max(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}})\boldsymbol{V} \tag{5}$$

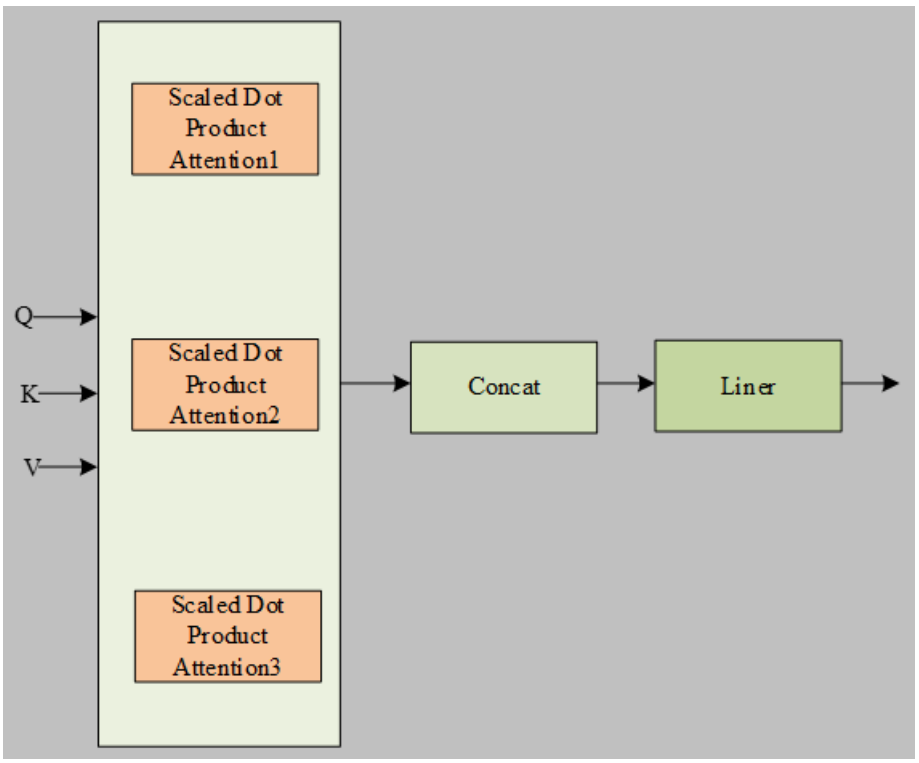
where, the query matrix, value matrix, and key matrix is denoted as $\boldsymbol{Q}$, $\boldsymbol{V}$, and $\boldsymbol{K}$ respectively, $\boldsymbol{W}^O$ represents the output matrix, $\boldsymbol{W}_i^Q$, $\boldsymbol{W}_i^K$, and $\boldsymbol{V}\boldsymbol{W}_i^V$ represent the matrices transformed by $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$, respectively; h represents the number of self-attentive heads. $MultiHead(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})$ indicates the output of multi-head attention module after transformation by multi-head information splicing, and its long-range feature capture capability is affected by the number of multi-head; the more the number, the better the feature capture effect.

For a given hot music $\{p_1,p_2,\&,p_M\}$, its vector representation is denoted as $\{e(p_1),e(p_2),\&,e(p_M)\}$; the current candidate music representation is x, the hot music features embedded representation $\boldsymbol{e}(\boldsymbol{p})$ is used as Query, and the candidate music representation $\boldsymbol{e}(\boldsymbol{x})$ is used as Key and Value. The features related to the hot music and the candidate music are represented as:

$$H = MultiHead(\boldsymbol{e}(\boldsymbol{p}),\boldsymbol{e}(\boldsymbol{x}),\boldsymbol{e}(\boldsymbol{x})) \tag{6}$$

In order to eliminate the inconsistency between hot music and candidate music, an average pooling operation is performed to obtain the final hot music features:

Figure 5. Multi-Head self-attention mechanism



$$h = \frac{\sum_{j=1}^{n} H_j}{n} \tag{7}$$

where n represent the length of feature H.

The user interest extractor aims to learn user interest representation based on user click history. Similarly to hotspot information, users' interests are usually multifaceted. When considering whether a candidate music will be clicked or not by a user, they prefer to focus on the interests related to the current candidate music. To represent different interests of users, the multi-head attention mechanism is used for selecting click history related to the current candidate music. Specifically, for the ith clicked music $\boldsymbol{e}(C_i^u)$ of user u and the current candidate music x, the historical click music $\boldsymbol{e}(C_i^u)$ is used as Query, the candidate music representation $\boldsymbol{e}(\boldsymbol{x})$ is used as Key and Value. The features related to interest music and candidate music are represented as:

$$I_u = MultiHead(\boldsymbol{e}(C_i^u), \boldsymbol{e}(\boldsymbol{x}), \boldsymbol{e}(\boldsymbol{x})) \tag{8}$$

After average pooling, the final interest music features are obtained:

$$i_u = \frac{\sum_{j=1}^{m} I_u}{m} \tag{9}$$

where m represent the length of feature I_u.

Click Predictor

The purpose of click predictor is predicting a user's click probability score on each candidate music. Three vectors of candidate music representation, user interest representation and hotspot feature representation need to be considered. The following strategy is adopted in this model: first, the candidate music representation is stitched with the hotspot feature representation and the user interest representation, respectively, to obtain the hotspot-candidate vector and the user-candidate vector, which are calculated as shown in equations (10) and (11), respectively.

$$o_{h,x} = \tanh(w_1(h \oplus e(x)) + b_1) \tag{10}$$

$$o_{i_u,x} = \tanh(w_2(i_u \oplus e(x)) + b_2) \tag{11}$$

Then, the user's click probability score on that candidate music is calculated by the scoring module:

$$o_{i_u,h,x} = fuse(o_{i_u,x}, o_{h,x}) \tag{12}$$

$$\hat{y}_{u,x} = sigmoid(o_{i_u,h,x} + b\ ) \tag{13}$$

Finally, a sigmoid nonlinear variation is used as the activation function to forecast the probability of user's click for the current candidate of music.

Using the concatenation fusion method:

$$o_{i_u,h,x} = \tanh(w_3(o_{i_u,x} \oplus o_{h,x}) + b_3) \tag{14}$$

Loss Function

A negative log-likelihood function minimization approach is used to train the model:

$$L = -(\sum_{s \in S^+} y_{u,x} \log \hat{y}_{u,x} + \sum_{s \in S^-} (1 - y_{u,x}) \log(1 - \hat{y}_{u,x})) \tag{15}$$

where S- and S+ represent the negative and positive sample sets, respectively.

EXPERIMENT AND ANALYSIS

The experiments were performance by using a Lenovo laptop Legion R7000 2020 with an AMD Ryzen 5 4600H processor at 3.0 GHz and 64-bit Windows 10 operating system, as shown in Table 1.

Experimental Dataset

The FMA dataset is a dump from the publicly available Free Music Archive, which can be downloaded at https://github.com/mdeff/fma. All metadata for all tracks are in the zip file fma_metadata.zip. The content of tracks.csv is the metadata of all 106,574 tracks (Kouki et al., 2019). Fma_medium.zip contains 25,000 30-second tracks, and all tracks in fma_medium.zip are singularly divided into Hip-Hop, International, Electronic, Folk, Experimental, Rock, and Instrumental. Each track in each category is a 30-second clip in mp3 format.

Unlike the FMA dataset, the GTZAN dataset (Chen et al., 2018) has 10 categories, each of which includes multiple music clips at 22050Hz approximately 30 seconds in length and in mp3 format for a total of 10,000 tracks in the categories Popular, Reggae, and Rock in 10 categories, and it is available for download at https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification.

Evaluation Index

The model uses the average precision (AP) as the accuracy evaluation index, and the larger the AP value which proves the model has better detection effect. The MAE (mean absolute deviation) and root mean square error (RMSE) are displayed as evaluation indexes:

Table 1. Experimental platform settings

Experimental environment	Specific information
Operating system	Windows 10
Memory	64GB
Language	Python3.6
Development tool	Pycharm
Raphics card	NVIDIA GeForce GTX 1060 6G
Development platform	Tensorflow2.0.0-beta
CPU	AMD Ryzen 5 4600H

Table 2. FMA data set distribution

Dataset	Training set	Validation set	Test set	Total number
FMA	15000	5000	5000	25000

Table 3. GTZAN data set distribution

Dataset	Training set	Validation set	Test set	Total number
GTZAN	8000	1000	1000	10000

$$P = \frac{TP}{TP + FP} \tag{16}$$

$$R = \frac{TP}{TP + FN} \tag{17}$$

$$AP = \int_0^1 PRdR \tag{18}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (\hat{r}_i - r_i)}{N}} \tag{19}$$

$$MAE = \frac{\sum_{i=1}^{N} | \hat{r}_i - r_i |}{N} \tag{20}$$

TP represents the number of positive samples that are identified correctly. FP represents the account of negative samples that are misreported. In addition, TN represents the number of negative samples correctly identified, and FN represents the number of positive samples that are missed.

Model Training

Loss Value During Training

The deep learning based personalized music recommendation method constructed in this paper was trained on the previously mentioned dataset. The process of the loss values obtained during the training and testing of 700 epochs is shown in Figure 6. The variation of the loss rate on the FMA and GTZAN datasets is given in Figure 6. After 100 and 200 iterations, the proposed model basically reaches convergence on the GTZAN dataset and the FMA dataset, with the loss rates remaining around 0.103 and 0.112.

Dropout Training

The effects of dropout values of 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9 on the accuracy of the model were tested on the FMA and GTZAN datasets, and the results of the test are illustrated in Figures 7 and 8. From the test set's results in the figures, when the loss rate is less than 0.7, the model appears to be over-fitted under both datasets; when the random loss rate is over 0.7, the model appears to be under-fitted. Therefore, the random loss rate of 0.7 is chosen to optimize the model in the experiments of this paper.

Experimental Comparison

Performance Comparison of Different Models

Firstly, the MAE results of the BTSAMA method suggested in this paper and the MCRN (Mao et al., 2022), GASM (Weng et al., 2022), and IR-MC (Tofani et al., 2022) methods were analyzed by

Figure 6. Loss value during training

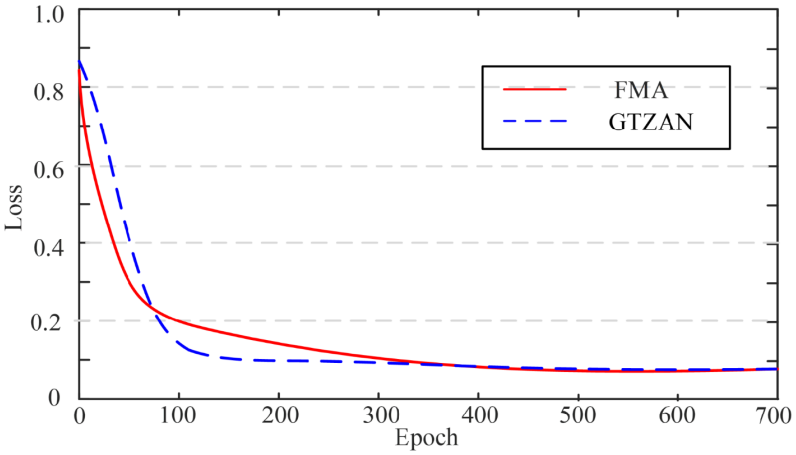
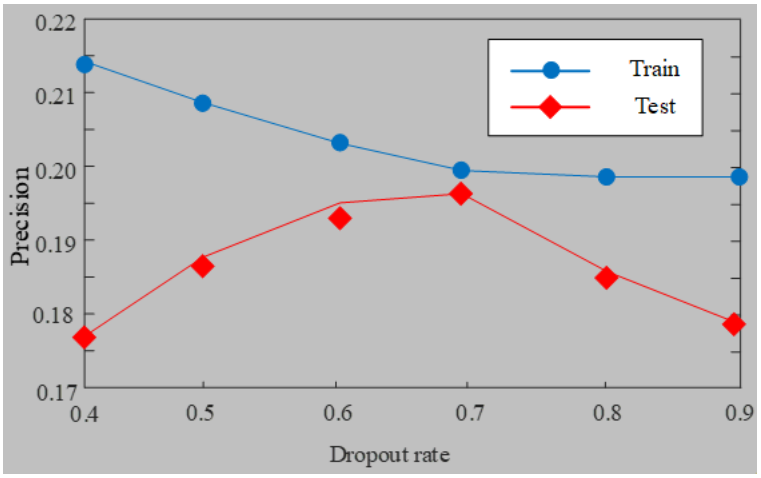
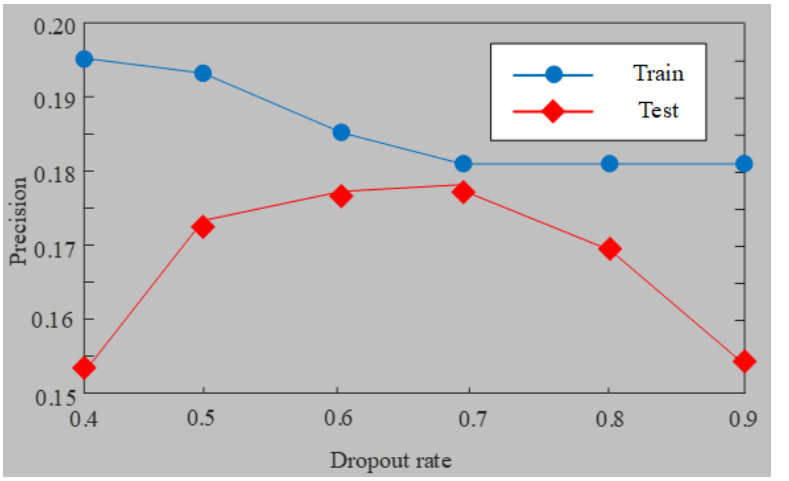


Figure 7. Effect of dropout on model results under FMA data set



experimental comparisons on two datasets, FMA and GTZAN, and their experimental comparisons are displayed in Table 4 and Figure 9. It is obvious that the BTSAMA model performs significantly better than the other comparison models, and the MAE result values of the BTSAMA model on the FMA and GTZAN datasets are 0.156 and 0.146, respectively, an improvement of at least 6.6% and 3.3% compared to other comparable models. The reason is that the TextCNN model and BERT model are conflated by BTSAMA to acquire classification of music reviews, specifies the review text and captures local music continuous features, introduces self-attention to address the remaining omitted non-continuous features that are not focused on by TextCNN, and combines the fused sequences to obtain the user's interest through a multi-headed attention mechanism network vector, abstracting deeper contextual internal semantic associations and improving the personalized music recommendation performance of the model. In contrast to several other models, BTSAMA all acquire the best results. Analyzing the reasons, it is obvious that MCRN and GASM do not consider the attention mechanism to get deeper intra-contextual semantic associations, and they do not consider the data imbalance problem, which affects the final experimental results.

Figure 8. Effect of dropout on model results under GTZAN dataset



Similarly to MAE, RMSE is also a common metric for model performance evaluation, except that MAE reflects the true error, while RMSE is the sum of squared errors before squaring, which amplifies the impact of errors and is more sensitive to outliers. In general, the MAE results are expected to be slightly smaller than the RMSE, which is essential for the objective evaluation of the model. The comparison of RMSE results for the two datasets is listed in Table 5. In order to see more clearly the performance of diverse models on both FMA and GTZAN datasets, the data in Table 5 are graphically presented, as shown in Figure 10: the horizontal axis of the bar chart represents the MCRN, GASM, IR-MC, and BTSAMA models, respectively, and the vertical axis is the RMSE result values of each model on the two datasets of FM and GTZAN, respectively. Their heights directly reflect the good or bad model performance.

In Table 5 and Figure 10, the RMSE values of the BTSAMA model on the FMA and GTZAN data sets are 0.185 and 0.164, respectively, an improvement of at least 12.4% and 5.2% compared to other comparable models. Combining with the above analysis shows that the BTSAMA model has a greater advantage in performance. This is because the proposed method achieves the classification of music reviews, the specification of music review texts, and the capture of continuous features of local music, and through the effective combination of both, the feature information contributing to music recommendation is obtained, which effectively enhances the performance of the personalized music recommendation model.

Effect of Recommended Music k on the Performance of Different Models

Aiming to demonstrate the merits of the proposed personalized music recommendation method, comparing different recommendation methods and BTSAMA under the same experimental conditions

Table 4. Comparison of MAE results for the two datasets

Model	FMA	GTZAN
MCRN	0.235	0.230
GASM	0.212	0.205
IR-MC	0.167	0.151
BTSAMA (proposed)	0.156	0.146

Figure 9. Comparison of MAE of two datasets under different methods

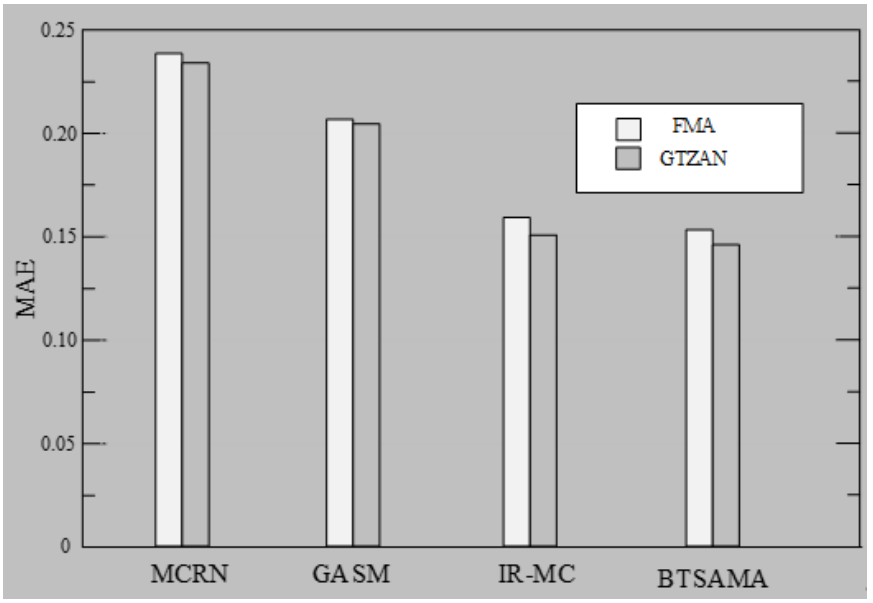
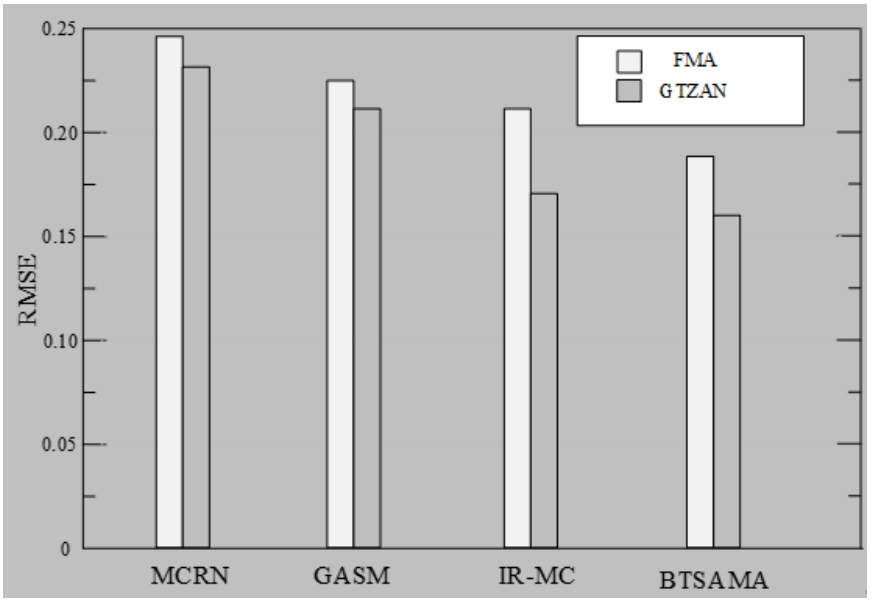


Figure 10. Comparison of RMSE of two datasets under different methods



with k=10, 15, 20, and 25 and the three dimensions of Precision, Recall, and MAP were used to measure the effectiveness of each model for personalized music recommendation, and the obtained results are displayed in Figure 11~ Figure 16.

When k=10, the accuracy, average accuracy values, and recall of the proposed BTSAMA are 0.193, 0.231, and 0.112, for the FMA dataset, and 0.181, 0.103, and 0.215 for the GTZAN dataset, separately; these are better when compared to the other models. The reason is that BTSAMA effectively

Table 5. Comparison of RMSE results for the two datasets

Model	FMA	GTZAN
MCRN	0.247	0.238
GASM	0.225	0.212
IR-MC	0.208	0.173
BTSAMA (proposed)	0.185	0.164

Figure 11. Precision comparison under FMA dataset

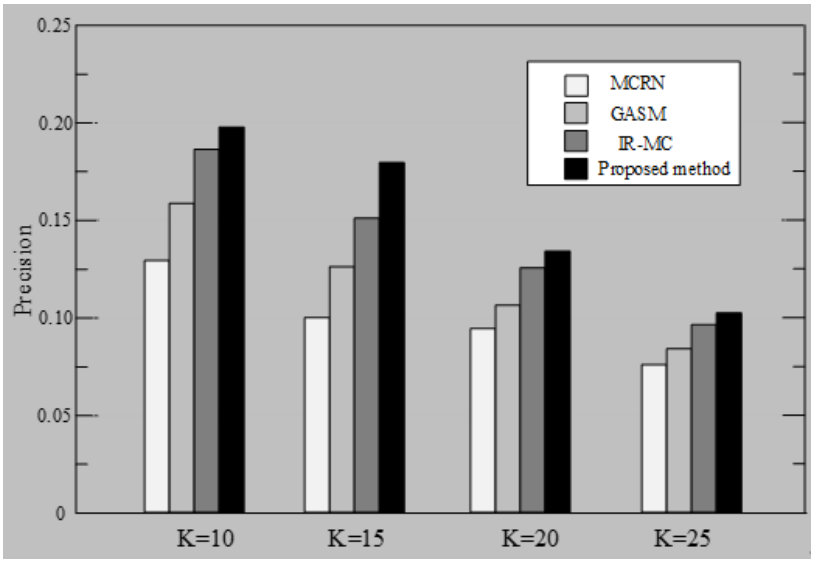


Figure 12. Recall comparison under FMA dataset

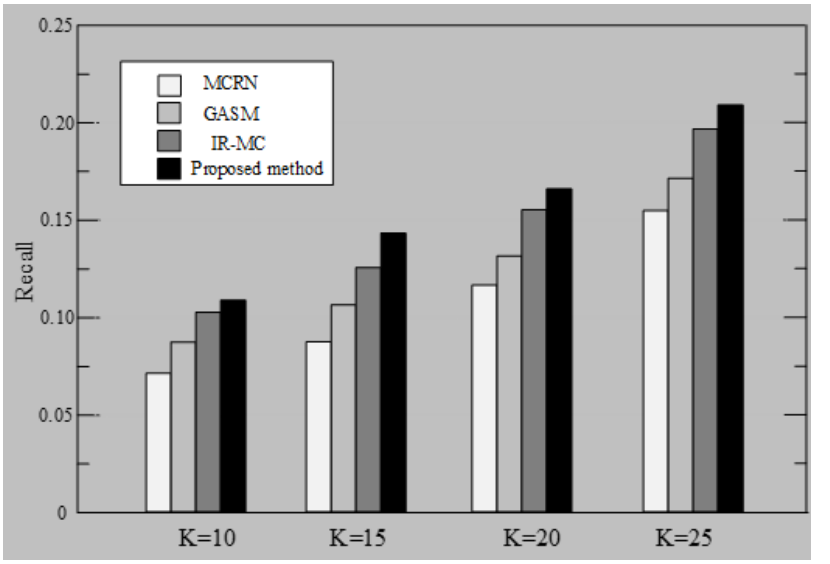


Figure 13. MAP Comparison under FMA dataset

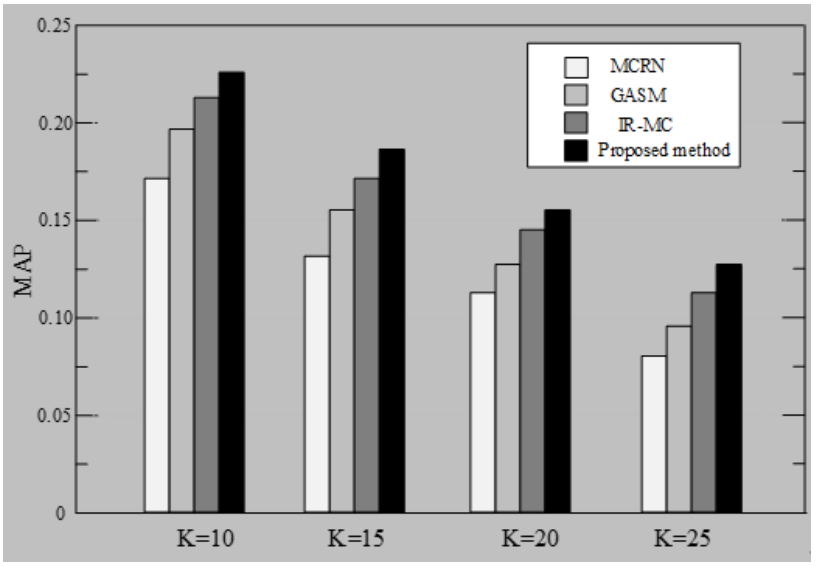
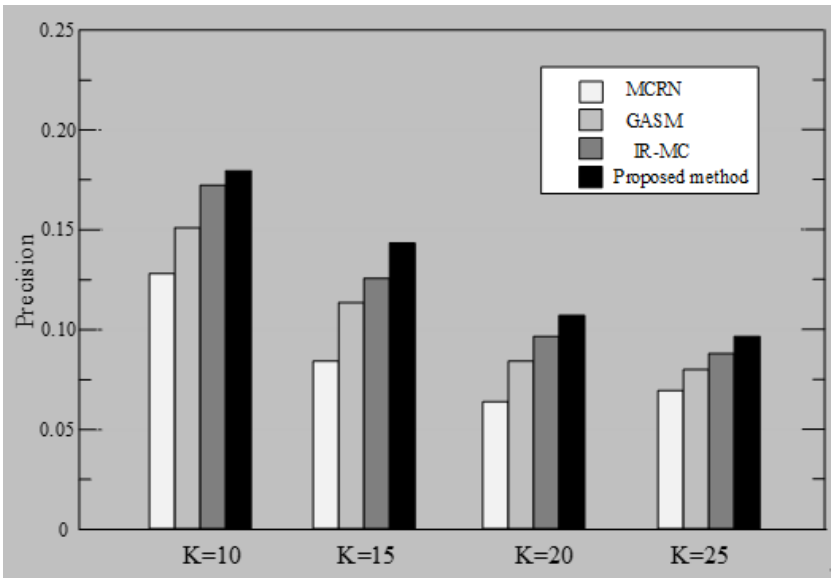


Figure 14. Precision comparison under GTZAN dataset



improves the text feature extraction ability in the personalized music recommendation task by fusing left and right contextual information through the BERT pre-training model and using TextCNN to acquire the sequence feature matrix.

Ablation Experiment

In this paper, the corresponding ablation experiments were designed using the controlled variable method.

Figure 15. Recall comparison under GTZAN dataset

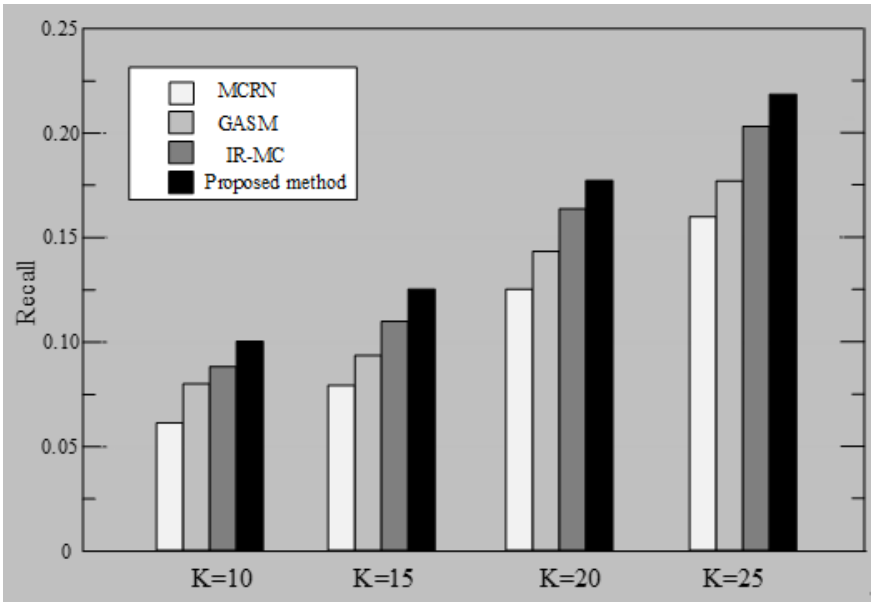
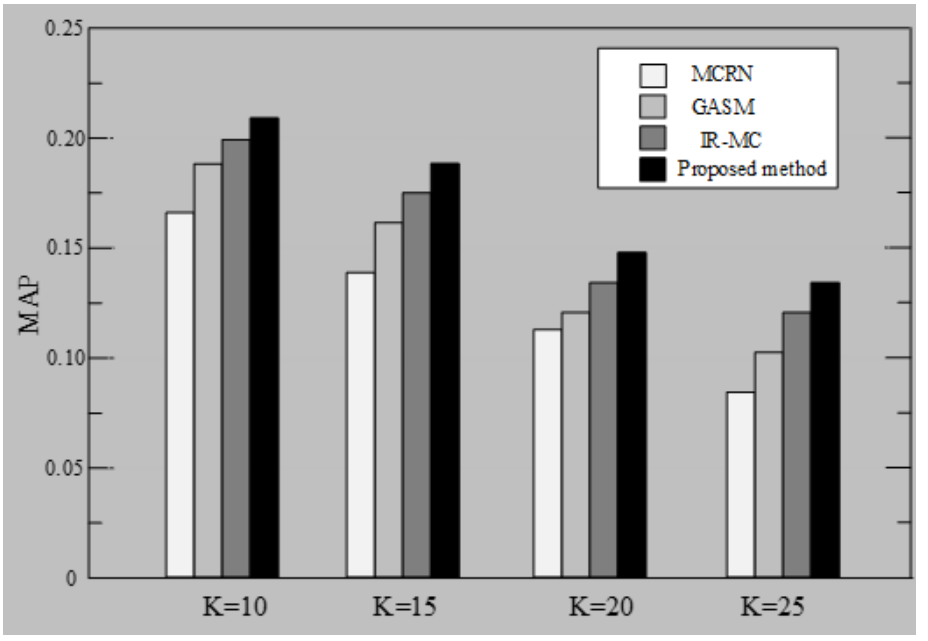


Figure 16. MAP comparison under GTZAN dataset



1. Experiment I: TextCNN;
2. Experiment II: BERT-TextCNN (BT) method;
3. Experiment III: BERT-TextCNN-Self-Attention (BTSA) method;
4. Experiment IV: BERT-TextCNN-Self-Attention-Muti-head-Attention (BTSAMA) method

The results of experiment prove that the MAE and RMSE metrics of BTSAMA are optimal under the FMA and GTZAN datasets. The reason is the BTSAMA builds the language model using BERT pre-trained word vectors and dynamically generates semantic vectors based on the word context to capture semantic information through TextCNN. In this paper, a multi-headed attention mechanism is selected to abstract deeper contextual internal semantic associations, which improves the performance of the model for personalized music recommendation.

CONCLUSION

A personalized music recommendation method combining TextCNN and attention is proposed. The BERT model is combined with and TextCNN model so as to acquire the music review's classification and specification of review text and to capture local music continuous features, introduce self-attention to solve the rest of the omitted non-continuous features not paid attention to by TextCNN, combine the fused sequences to get the user's interest vector through a multi-headed attention mechanism network, abstract deeper of contextual internal semantic associations, which improves the model's performance for personalized music recommendation. Experimentally, the proposed BTSAMA is significantly better than the compared methods in personalized music recommendation.

The proposed recommendation model incorporates only music content, but it does not take into account other valid information. There are many kinds of music-related information (e.g., lyrics, genres, composers) available on music platforms, which can reflect the characteristics of music from different perspectives and help the model to comprehensively understand music and thus model users' behavioral preferences more accurately. Therefore, further work is needed to explore how to combine multiple auxiliary information to improve the capacity of the recommendation algorithm. User behavior toward music often occurs at specific times and scenarios, and the types of music that a user plays changes as the environment changes, so simply learning based on user-music interaction data alone can overlook a lot of important information. Therefore, subsequent work needs to integrate contextual information such as time and location to study user behavior preferences in specific contexts. In addition, the evaluation indicators MAE and RMSE selected in the experiments are very

Table 6. Results of ablation experiments on the FMA dataset

Model	MAE	RMSE
TextCNN	0.202	0.193
BT	0.178	0.174
BTSA	0.164	0.154
BTSAMA (proposed)	0.156	0.146

Table 7. Results of ablation experiments on the GTZAN dataset

Model	MAE	RMSE
TextCNN	0.226	0.206
BT	0.203	0.186
BTSA	0.192	0.172
BTSAMA (proposed)	0.185	0.164

suitable for application in large-scale music libraries and user group scenarios. However, this aspect of the experiment has not been executed yet. In the future, the proposed BTSAMA will be applied to larger music libraries and user groups to improve its scalability, and its predictive performance will be further improved through model optimization, parameter tuning, and other methods.

DATA AVAILABILITY

The data included in this paper are available without any restriction.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest to report regarding the present study.

FUNDING STATEMENT

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. Funding for this research was covered by the authors of the article

REFERENCES

Abdul, A., Chen, J., Liao, H. Y., & Chang, S.-H. (2018). An emotion-aware personalized music recommendation system using a convolutional neural networks approach. *Applied Sciences (Basel, Switzerland)*, *8*(7), 1103–1110. doi:10.3390/app8071103

Ayata, D., Yaslan, Y., & Kamasak, M. E. (2018). Emotion based music recommendation system using wearable physiological sensors. *IEEE Transactions on Consumer Electronics*, *64*(2), 196–203. doi:10.1109/TCE.2018.2844736

Bauer, C., & Schedl, M. (2019). Global and country-specific mainstreaminess measures: Definitions, analysis, and usage for improving personalized music recommendation systems. *PLoS One*, *14*(6), 17380–17389. doi:10.1371/journal.pone.0217389 PMID:31173583

Cai, X., Hu, Z., & Zhao, P. (2020). A hybrid recommendation system with many-objective evolutionary algorithm. *Expert Systems with Applications*, *15*(5), 1136–1145.

Chen, J., Zhuang, F., & Hong, X. (2018). Attention-driven factor model for explainable personalized recommendation. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, (pp. 909-912). ACM.

Chen, X., Li, S., & Li, H. (2019). Generative adversarial user model for reinforcement learning based recommendation system. *International Conference on Machine Learning PMLR*, (pp. 1052-1061). PMLR.

Drott, E. (2018). Why the next song matters: Streaming, recommendation, scarcity. *Twentieth-Century Music*, *15*(3), 325–357. doi:10.1017/S1478572218000245

Gunawan, A. A. S., & Suhartono, D. (2019). Music recommender system based on genre using convolutional recurrent neural networks. *Procedia Computer Science*, *15*(1), 99–109.

Guo, H., Tang, R., & Ye, Y. (2017). DeepFM: A factorization-machine based neural network for CTR prediction. arXiv preprint arXiv:1703.04247. 10.24963/ijcai.2017/239

Huang, L., Fu, M., Li, F., Qu, H., Liu, Y., & Chen, W. (2021). A deep reinforcement learning based long-term recommender system. *Knowledge-Based Systems*, *21*(3), 1067–1076. doi:10.1016/j.knosys.2020.106706

Jin, Y., Cai, W., & Chen, L. (2019). MusicBot: Evaluating critiquing-based music recommenders with conversational interaction. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, (pp. 951-960). ACM. doi:10.1145/3357384.3357923

Jin, Y., Tintarev, N., & Verbert, K. (2018). Effects of personal characteristics on music recommender systems with different levels of controllability. *Proceedings of the 12th ACM Conference on Recommender Systems*, (pp. 13-21). ACM. doi:10.1145/3240323.3240358

Karakayali, N., Kostem, B., & Galip, I. (2018). Recommendation systems as technologies of the self: Algorithmic control and the formation of music taste. *Theory, Culture & Society*, *35*(2), 3–24. doi:10.1177/0263276417722391

Kim, H. G., Kim, G. Y., & Kim, J. Y. (2019). Music recommendation system using human activity recognition from accelerometer data. *IEEE Transactions on Consumer Electronics*, *65*(3), 349–358. doi:10.1109/TCE.2019.2924177

Kim, M. S., & Kim, S. (2018). Factors influencing willingness to provide personal information for personalized recommendations. *Computers in Human Behavior*, *8*(2), 143–152. doi:10.1016/j.chb.2018.06.031

Kouki, P., Schaffer, J., & Pujara, J. (2019). Personalized explanations for hybrid recommender systems. *Proceedings of the 24th International Conference on Intelligent User Interfaces*, (pp. 379-390). ACM.

Kowald, D., Schedl, M., & Lex, E. (2020). The unfairness of popularity bias in music recommendation: A reproducibility study. *Advances in information retrieval: 42nd European conference on IR research,* (pp. 35-42). Springer.

Li, G., & Zhang, J. (2018). Music personalized recommendation system based on improved KNN algorithm. *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, (pp. 777-781). IEEE..

Li, Q., Myaeng, S., & Kim, B. (2007). A probabilistic music recommender considering user opinions and audio features. *Information Processing & Management*, *43*(2), 473–487. doi:10.1016/j.ipm.2006.07.005

Liebman, E., Saar-Tsechansky, M., & Stone, P. (2019). The right music at the right time: Adaptive personalized playlists based on sequence modeling. *Management Information Systems Quarterly*, *43*(3), 1–12. doi:10.25300/MISQ/2019/14750

LinZ.FengM.CiceroN. D. S.MoY.BingX.BowenZ.YoshuaB. (2017) A structured self-attentive sentence embedding [EB/OL]. (2017-03-09). https://arxiv.org/abs/1703.03130

Liu, S., Chen, Z., & Liu, H. (2019). User-video co-attention network for personalized micro-video recommendation. *The World Wide Web Conference*, (pp. 3020-3026). ACM. doi:10.1145/3308558.3313513

Mao, Y., Zhong, G., Wang, H., & Huang, K. (2022). Music-CRN: An efficient content-based music classification and recommendation network. *Cognitive Computation*, *14*(6), 2306–2316. doi:10.1007/s12559-022-10039-x

McInerney, J., Lacker, B., & Hansen, S. (2018). Explore, exploit, and explain: Personalizing explainable recommendations with bandits. *Proceedings of the 12th ACM Conference on Recommender Systems*, (pp. 31-39). ACM. doi:10.1145/3240323.3240354

Melchiorre, A. B., Rekabsaz, N., Parada-Cabaleiro, E., Brandl, S., Lesota, O., & Schedl, M. (2021). Investigating gender fairness of recommendation algorithms in the music domain. *Information Processing & Management*, *58*(5), 1026–1032. doi:10.1016/j.ipm.2021.102666

Millecamp, M., Htun, N. N., & Conati, C. (2019). To explain or not to explain: The effects of personal characteristics when explaining music recommendations. *Proceedings of the 24th International Conference on Intelligent User Interfaces*, (pp. 397-407). ACM. doi:10.1145/3301275.3302313

Millecamp, M., Htun, N. N., & Jin, Y. (2018). Controlling Spotify recommendations: Effects of personal characteristics on music recommender user interfaces. *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*, (pp. 101-109). ACM. doi:10.1145/3209219.3209223

Prey, R. (2018). Nothing personal: Algorithmic individuation on music streaming platforms. *Media Culture & Society*, *40*(7), 1086–1100. doi:10.1177/0163443717745147 PMID:30270951

Sachdeva, N., Gupta, K., & Pudi, V. (2018). Attentive neural architecture incorporating song features for music recommendation. *Proceedings of the 12th ACM Conference on Recommender Systems*, (pp. 417-421). ACM. doi:10.1145/3240323.3240397

St. Garcia-Gathright, J., Thomas, B., & Hosey, C. (2018). Understanding and evaluating user satisfaction with music discovery. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, (pp. 55-64).

Tofani, A., Borges, R., & Queiroz, M. (2022). Dynamic session-based music recommendation using information retrieval techniques. *User Modeling and User-Adapted Interaction*, *32*(4), 575–609. doi:10.1007/s11257-022-09343-w

Wang, D., Deng, S., & Xu, G. (2018). Sequence-based context-aware music recommendation. *Information Retrieval*, *2*(3), 230–252. doi:10.1007/s10791-017-9317-7

Wang, R., Ma, X., Jiang, C., Ye, Y., & Zhang, Y. (2020). Heterogeneous information network-based music recommendation system in mobile networks. *Computer Communications*, *15*(1), 429–437. doi:10.1016/j.comcom.2019.12.002

Wei, J., He, J., Chen, K., Zhou, Y., & Tang, Z. (2017). Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems with Applications*, *6*(9), 29–39. doi:10.1016/j.eswa.2016.09.040

Wei, Y., Wang, X., & Nie, L. (2019). MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. *Proceedings of the 27th ACM International Conference on Multimedia*, (pp. 1437-1445). ACM. doi:10.1145/3343031.3351034

Weng, H., Chen, J., Wang, D., Zhang, X., & Yu, D. (2022). Graph-based attentive sequential model with metadata for music recommendation. *IEEE Access : Practical Innovations, Open Solutions*, *10*(4), 108226–108240. doi:10.1109/ACCESS.2022.3213812

Werner, A. (2020). Organizing music, organizing gender: Algorithmic culture and Spotify recommendations. *Popular Communication*, *18*(1), 78–90. doi:10.1080/15405702.2020.1715980

Zhang, W., Zhang, X., Wang, H., & Chen, D. (2019). A deep variational matrix factorization method for recommendation on large scale sparse dataset. *Neurocomputing*, *33*(2), 206–218. doi:10.1016/j.neucom.2019.01.028

Zhang, Y., Yin, C., Wu, Q., He, Q., & Zhu, H. (2019). Location-aware deep collaborative filtering for service recommendation. *IEEE Transactions on Systems, Man, and Cybernetics. Systems*, *51*(6), 3796–3807. doi:10.1109/TSMC.2019.2931723

Zhao, G., Fu, H., Song, R., Sakai, T., Chen, Z., Xie, X., & Qian, X. (2019). Personalized reason generation for explainable song recommendation. [TIST]. *ACM Transactions on Intelligent Systems and Technology*, *10*(4), 1–21. doi:10.1145/3337967

Zhao, G., Lou, P., Qian, X., & Hou, X. (2020). Personalized location recommendation by fusing sentimental and spatial context. *Knowledge-Based Systems*, *19*(2), 1058–1065. doi:10.1016/j.knosys.2020.105849

Zhao, X., Xia, L., & Zhang, L. (2018). Deep reinforcement learning for page-wise recommendations. *Proceedings of the 12th ACM Conference on Recommender Systems*, (pp. 95-103). ACM. doi:10.1145/3240323.3240374

Zheng, E., Kondo, G. Y., Zilora, S., & Yu, Q. (2018). Tag-aware dynamic music recommendation. *Expert Systems with Applications*, *10*(3), 244–251. doi:10.1016/j.eswa.2018.04.014

Shaomin Lv, born in Taoyuan, Taiwan Province in 1985, obtained a doctoral degree in Musical Arts from Michigan State University. Now he teaches in Yulin Normal University and is engaged in the teaching and training of music performance, piano, wind orchestra training and other courses.