Music Emotion Recognition based on Segment-level Two-stage Learning

Na He[1*] and Sam Ferguson[1]

[1]School of Computer Science, Faculty of Engineering & IT, University of Technology Sydney, NSW, 2007, Australia.

*Corresponding author(s). E-mail(s): winterhn@gmail.com;
Contributing authors: samuel.ferguson@uts.edu.au;

Abstract

In most Music Emotion Recognition (MER) tasks, researchers tend to use supervised learning models based on music features and corresponding annotation. However, few researchers have considered applying unsupervised learning approaches to labelled data except for feature representation. In this paper, we propose a segment-based two-stage model combining unsupervised learning and supervised learning. In the first stage, we split each music excerpt into contiguous segments and then utilize an autoencoder to generate segment-level feature representation. In the second stage, we feed these time-series music segments to a Bidirectional Long Short-Term Memory deep learning model to achieve the final music emotion classification. Compared with the whole music excerpts, segments as model inputs could be the proper granularity for model training and augment the scale of training samples to reduce the risk of overfitting during deep learning. Apart from that, we also apply frequency and time masking to segment-level inputs in the unsupervised learning part to enhance training performance. We evaluate our model on two datasets. The results show that our model outperforms state-of-the-art models, some of which even use multimodal architectures. And the performance comparison also evidences the effectiveness of audio segmentation and the autoencoder with masking in an unsupervised way.

Keywords: segment-level representation, unsupervised learning, music emotion recognition, autoencoder

1 Introduction

Within the research area of Music Information Retrieval (MIR), emotion recognition is an important branch and benefits various MER application areas. In recent years, deep learning models have become primary methods used to implement emotion prediction [1, 2]. With layers of neural networks, these models are capable of learning music features automatically from raw audio or low-level audio features. In Music Emotion Recognition (MER) tasks, much research is based on music datasets containing emotion annotation, which naturally adopts supervised learning methods to find patterns between each music input and its corresponding annotation. Few studies take into account unsupervised learning for labelled data.

In addition, most researchers keep the duration of each audio input in accordance with the given annotation, seldom considering the effect of changing that duration. For dynamic emotion detection, to match the time-varying annotation sampling frequency which is usually 2 Hz or 1 Hz, the length of each music clip is 0.5s or 1s. These audio clips are fed into a training model [3] and thus implement a one-to-one mapping with those

labels. For static emotion recognition, each music excerpt (usually the duration of 30s or more) corresponds to one annotation. According to this approach, researchers usually extract music features from these music excerpts without further splitting them into shorter segments. However, not all music duration are appropriate for emotion analysis and model training [4, 5]. Some research even splits longer-duration music recordings into a series of short segments but assign presumptive segment-level labels as the training targets rather than using the original annotation [6]. Few research has paid attention to adjusting the length of audio input without adding extra annotation.

In this paper, we focus on static emotion recognition and propose an architecture that uses music segments split from each music excerpt as model inputs, while only using the original emotion annotation. Here we divide our framework into 2 parts. The first part is an unsupervised learning model which generates the feature representation for segment-level music without defining new emotion labels for them. The second part is a supervised training model where we view segments as the sequential units of each music excerpt and train them in a deep learning model of handling time-series data to predict the final emotion. In the module of unsupervised learning, we utilize the *SpecAugment* technique [7] to partially mask log-mel spectrogram input data from frequency and time dimensions to enhance the robustness of the training model.

The main contribution of this work is designing a two-stage MER architecture that combines segment-based unsupervised learning as a feature extractor and supervised learning as an emotion detector. In this way, we could split each music excerpt into contiguous segments without having to provide segment-level annotations, and feed them into appropriate training models to explore potential features effectively. From the perspective of data augmentation, segment-level music with partial masking increases the data scale and data variation for unsupervised learning, thereby boosting the model performance.

2 Related works

With the evolution of MIR research, deep learning has played a vital role in improving performance. Based on such models, various factors have been considered, including feature sources, feature representation and model design.

2.1 Feature Source

To train a deep learning model, the first thing researchers need to determine is what kind of sources are used to extract features. Music audio data is the primary consideration. Compared with traditional machine learning where tens or hundreds of human-engineered features are selected, the typical inputs for deep neural networks are 1-dimensional (1D) raw audio data [8], 2-dimensional (2D) mel-scaled spectrogram [9] or a mix of both [10]. Further, some research made use of a Music source separation (MSS) module *Demucs* [11] to generate vocals, drums, bass and other sources from the raw waveform and fed them into deep learning models with three fusion strategies [12]. On the other hand, some attempts have been made only using lyrics [13, 14] or electroencephalogram (EEG) signals [15]. Apart from this, much research tends to employ multimodal methodologies based on multiple data sources to take advantage of their complementarity. Among them, the combination of audio and lyrics is a popular solution [16, 17]. In some cases, researchers achieved better performance by leveraging audio as the main source and aggregating supplementary resources such as Electrodermal Activity (EDA) [18], social tags [19] and even facial expression images when the video is available [20].

2.2 Feature Representation

In recent years, feature representation has gained more attention in many studies on account of training deep neural networks more efficiently. Distinct from engineered features extracted from source data directly, feature representation benefits from the ability of deep learning to extract more meaningful information and generate vector-based features to represent sources. One practical method is utilizing unsupervised learning models. Generally, unsupervised learning is used to analyze unlabelled datasets for the purpose of clustering, association, and dimensionality reduction. In recent years, with the development of neural networks, unsupervised learning models could learn efficient feature representation from data input, such as an autoencoder or Restricted Boltzmann

Machine (RBM) [21]. In this situation, unsupervised learning models usually act as feature extractors, followed by supervised learning models for prediction. Sometimes, unsupervised learning is also used to correlate and blend the multimodal features into new features that contain more common information [22]. Furthermore, transfer learning is another well-known approach for feature representation. Fan et al. [23] utilized a pretrained model *VGGish* [24] as a feature extractor where the audio data is converted into latent feature vectors as inputs for subsequent training. MusiCoder [25] combined these two approaches. They conducted unsupervised learning on unlabeled audio data to build up a pretrained model which serves other labelled datasets to form feature representation. In our work, we adopt unsupervised learning to extract feature representations.

2.3 Model Design

To pursue better performance, researchers have put a great deal of effort into investigating alternative model designs. Inspired by the success of deep learning in image detection, convolutional neural network (CNN) models are applied widely in MIR research [6, 8, 26]. Such models could exploit highly-abstract features automatically from inputs. Since music is sequential data, recurrent neural network (RNN) models have become a complementary approach for capturing time-varying information [10, 17, 27]. These models were ever used to make up the unsupervised autoencoder [28]. Besides CNNs and RNNs, multiple layers of multi-head attention model was proposed as the components of an autoencoder for music classification [25], which is also known as the transformer architecture inspired by research in Natural Language Processing (NLP) [29] and speech recognition [30]. However, the complexity of this approach is very high and the pre-training duration is beyond 800 hours for each dataset. It may not productive for some MER tasks to train such attention model in terms of computing cost. Regarding multimodal fusion, deep learning also contributes to fusion strategies. An emerging strategy takes advantage of graph attention networks (GAT) to make decision-level fusion [31], and could be a good option for future research.

As mentioned above, most research work concentrates on model design regardless of the impact of the length of each input. Focusing on audio segmentation, Wu et al [32] argued that song-level features may lead to inaccurate feature representation for emotion recognition due to music emotion varying between segments. However, emotion is mostly consistent within each segment. Further, Aljanaki et al [33] distinguished emotional segments from structural segments for music. They compared these two types of segmentation, and found that emotional boundaries coincide with structural boundaries very often. Therefore, segment-level emotion detection for music is reasonable. In practice, Lee et al [8] compared a sample-level deep learning approach with a frame-level approach through configuring convolutional filter length and stride rather than partitioning the raw waveform directly. The segmentation occurs during training, which leads to no way to obtain segment-level data for additional manipulation. In contrast, Sarkar et al [6] split each audio clip into 5-second segments and transformed them into mel-scaled spectrogram as inputs to a *VGGNet*-style model. But they assigned clip-level labels to segments as training targets, which may mislead the final prediction. In our work, we are allowed to process segment-level data before training so as to find more ways to improve performance. Meanwhile, no extra labels are required for segments.

3 Methodologies

We propose a two-stage learning framework as seen in Fig. 1. The first stage is an unsupervised learning model to obtain segment-level feature representation. The second stage is a supervised learning model to predict emotion classification. For feature source, we use music audio data to serve this model structure. For emotion taxonomy, we follow 2D valence-arousal space initiated by Russell [34] and view it as a classification problem.

3.1 Feature Representation

The detailed design for feature representation is shown in Fig. 2. In this part, we first process the audio data and transform it into log-mel spectrogram. Then we partially mask these data from time and frequency dimensions separately. After that, these data are passed into an autoencoder architecture to encode and decode with the target

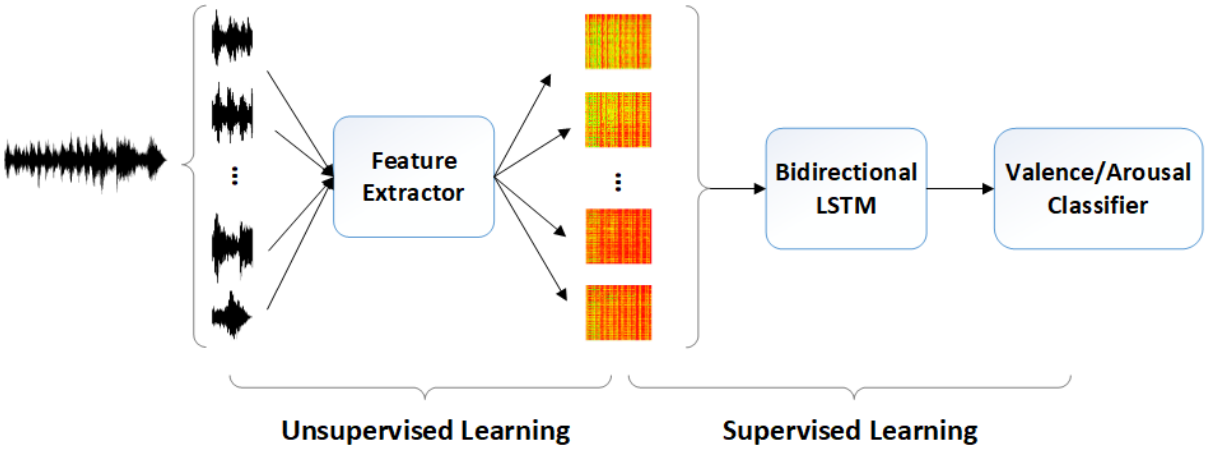


Fig. 1 Model overview: The two-stage learning framework includes an unsupervised learning model as a segment-level feature extractor and a supervised learning model as an emotion recognizer

of minimizing the loss between the reconstructed outputs and the inputs. In this way, the feature encoder module with the optimized training weights become a feature extractor which accepts log-mel spectrogram of segment-level audio data and outputs their feature representation.

3.1.1 Frequency and Time Masking

Inspired by SpecAugment [7] and MusiCoder [25], we mask the input data partially to increase the robustness of the training model against partial loss of information. More importantly, this procedure feeds the model with deliberately perturbed data to reduce overfitting during training. Due to the log-mel spectrogram applied, we mask such data in both the frequency domain and time domain.

Frequency masking: Given the total number of mel frequency channels F_c, we set the frequency mask parameter F and make $F < F_c$. We specify a span of consecutive mel frequency channels $[f_0, f_0 + f)$ to be masked, where f is a randomly selected number from a uniform distribution over $[0, F)$ and f_0 is a randomly selected number from a uniform distribution over $[0, F_c - f)$.

Time masking: Given a log-mel spectrogram with the total time steps T_s, we set the time mask parameter T and make $T < T_s$. We specify a span of consecutive time steps $[t_0, t_0 + t)$ to be masked, where t is the randomly selected number from a uniform distribution over $[0, T)$ and t_0 is the randomly selected number from a uniform distribution over $[0, T_s - t)$.

Here we mask one span for each domain. Because the time duration for each segment is not very long and only mel-scaled frequency is included. Masking multiple spans of time or frequency may increase the risk of under fitting during training due to too much information loss. For the option of masked value that replace true value, either zero or the mean value could be applied. We compare these two situations in our experiments to find the best performance.

3.1.2 Convolutional Autoencoder

As shown in Fig. 2, this autoencoder model is a deep CNN-based architecture and consists of a feature encoding module and a decoding module. We feed the masked log-mel spectrogram data into the feature encoder and train the whole autoencoder model. Once the output of the decoder achieves the minimized loss against the original input, we save the optimized weights for the feature encoder that is used as a feature extractor to generate latent feature representation. The feature encoder consists of 3 groups of stacked layers where each 2D CNN layer is followed by a 2D max-pooling layer. The CNN layers extract latent audio features and the max-pooling layers compact representations. The output of the feature encoder retains the most relevant information of the input and achieves dimensionality reduction, while the reconstruction work is implemented by the decoder where a series of 2D CNN layers with 2D upsampling layers are applied. Here the

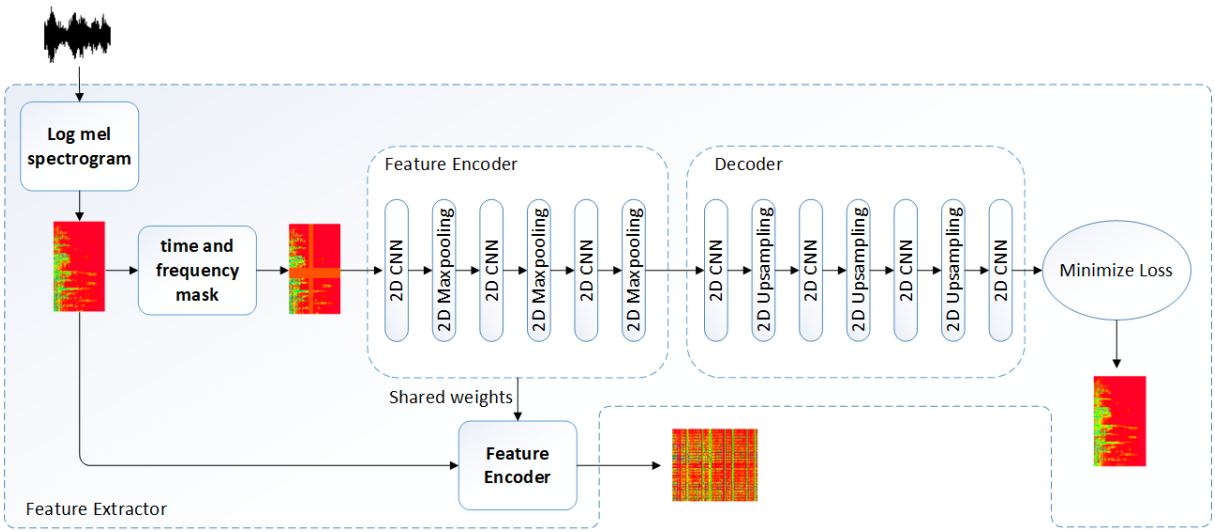


Fig. 2 The detailed design for feature representation. For each segment-level audio, it is transformed into log-mel spectrogram, followed by frequency and time masking. Then such input is fed into a CNN-based autoencoder with the target of minimizing loss. The feature encoder with the optimized weights is used as a feature extractor to provide segment-level feature representations

2D CNN layers perform deconvolution and cooperate with the upsampling layers to reconstruct the original data. For each CNN layer, the rectified linear units (ReLU) activation function is used to improve training efficiency. Through this unsupervised learning architecture, we extract feature representations for music segments without labelling the emotion for them.

3.1.3 Loss Function

During the autoencoder model training, we monitor the best reconstruction driven by minimizing Huber loss. Huber loss is a robust regression loss that is less sensitive to outliers than the squared error loss [35]. This loss function is defined as below,

$$L_\delta(x) = \begin{cases} 0.5 \cdot x^2 & if\ |x| \leqslant \delta \\ \delta \cdot |x| - 0.5 \cdot \delta^2 & \text{otherwise} \end{cases} \quad (1)$$

where x means the difference between the observed and predicted values. We set $\delta = 1$ by default. In this way, Huber loss could reduce the impact of the outliers and promote training convergence [25].

3.2 Emotion Classification

The second part of our framework is a supervised learning structure for emotion classification. A Bidirectional Long Short-Term Memory (BiLSTM) model is utilized to capture temporal music information and detect emotion classification. For this model, each input is a sequence of feature representations of time-series segments which constitute one music excerpt. The output is the Valence/Arousal (VA) predictions corresponding to this music excerpt. From the perspective of model implementation, we can regard the feature encoder and BiLSTM as a whole. During training, the encoder module is frozen and holds the optimal weights from unsupervised training while the BiLSTM neural network tunes the weight itself to achieve the final fitting.

4 Experiment

4.1 Dataset Description

To validate the model, we employ the PMEmo dataset[1], which is designed for MER research. The dataset contains songs with VA annotations, song metadata, EDA signals, pre-computed audio features, lyrics and even user comments. This music

[1] https://github.com/HuiZhangDB/PMEmo

set targets popular songs and selects the chorus part for each song in mp3 format. Among the total 794 songs, we select 767 songs that have been labelled with static VA annotations. Regarding annotation consistency, each subject listened to 20 excerpts including duplicated ones. Each song was annotated by at least 10 subjects and the bias for repeated annotation from one subject was taken into consideration. So that the quality of the annotation is guaranteed. The chorus excerpts are of various length and most of them are not less than 30 seconds (30s). According to this, we retain 30s for each song. For songs less than 30s, we pad them to 30s by repeating themselves from the start to the end. Totally, 230 clips are processed. In this manner, we make sure all music excerpts are the same duration to facilitate subsequent audio processing. More details about this dataset could refer to PMEmo document [36]. Based on this dataset, we compare our model with previous models to check the effect of audio segmentation and model architecture. However, PMEmo dataset has some problems such as single genre and imbalanced target labels. It is necessary to add another dataset to support some viewpoints in our experiment.

To prove the effectiveness of our model, we also validate our model on AllMusic dataset [37]. This dataset contains 900 song clips balanced in terms of Russell's VA quadrants and genres in each quadrant, which avoids the pitfall of PMEmo dataset. The quadrantal annotation is obtained based on AllMusic emotion tags and Warriner's list [38]. A manual blind inspection was conducted to exclude songs with unclear emotions so as to validate the annotation. Most songs are 30-second clips. Only about 2% songs need be padded to 30s by using the same strategy in PMEmo dataset. This dataset is mainly used to check the performance of different segment duration and masking.

4.2 Audio Processing

We process this music audio data to prepare the inputs for the training model. First, we split each 30-second music excerpt into contiguous segments. The selection of the segment duration should balance the validity of emotional response and the homogeneity of each segment for feature learning, and meanwhile consider the model adaptability. Referring to previous research [4, 23, 39, 40], we test segment duration from the value set of {1s, 3s, 5s, 10s} and compare the results. For PMEmo dataset, due to audio signal values falling into the range $[-1, 1]$, no extra normalization is required. For AllMusic dataset, we normalize data into the same range.

We then convert each segment-level audio into a mel-scaled spectrogram S_m by using the function provided in Python Librosa[2] package. To reduce the impact of outliers, S_m is further transformed into logarithmic scale base 10. The detail is defined as below:

$$S_{lm} = lg(S_m + \Delta) \tag{2}$$

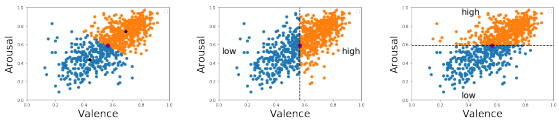
where we set Δ as 1 rather than a tiny increment like $1e-6$. In the preliminary experiment, we found that $\Delta = 1$ could result in relatively narrow data range with non-negative numbers, which bring about lower reconstruction losses. After that, we transpose 2D log-mel spectrogram data to generate the inputs before the masking operation. The expected data size for each input is 216×128, where 128 represents the number of mel-frequency channels while 216 is the number of fast Fourier transform (FFT) windows calculated from audio data. In order to gain the same data shape for different segment duration to adapt to the model, we need to adjust the length of the FFT window *n_fft* and the number of samples between successive windows *hop_length* when computing the mel spectrogram. Table 1 lists the parameters for mel spectrogram transformation.

Table 1 The parameters for mel spectrogram transformation

Dataset	Sample Rate	Segment Duration	n_fft	hop_length
PMEmo	44100Hz	1s	1024	205
		3s	1024	615
		5s	2048	1024
		10s	2048	2048
AllMusic	22050Hz	3s	1024	307
		5s	1024	512
		10s	2048	1024

In this table, 's' denotes second. For AllMusic dataset, '1s' segment duration is inapplicable due to the limitation of the model input shape.

[2] https://librosa.github.io/librosa/



(a) (b) (c)

Fig. 3 The distribution of static emotion annotation and the division for target classes. (a) 2 centers of k-means clustering and their midpoint (b) binary classification for high/low valence (c) binary classification for high/low arousal

For the frequency and time masking, we set $F = 30$ and $T = 32$. Then we pad the masked spans by either zero or the mean value of log-mel spectrogram. As observed, the mean value is not zero but the gap is small. Hence, padding the mean value shows a very small increase in performance. In the following experiments, we use the mean value to mask the frequency and time spans.

4.3 Annotation Transformation

For PMEmo dataset, the original annotation data was based on subjective responses in the range from 1 to 9 for both valence and arousal, and was scaled into $[0, 1]$ in the form of continuous values for storage in the dataset. To consider this task as a classification problem, we need to transform these continuous values into categories. We observe the distribution of the annotation data in 2D emotion space as seen in Fig. 3. Quadrantal classification is not appropriate due to imbalanced training samples in each quadrant. Thus, we use binary classification based on high/low level for each dimension. Moreover, we draw on the method used in [18, 41] to adjust the neutral threshold. That is, K-means clustering is applied to generate 2 clusters, followed by calculating 2 cluster centers and their midpoint. Then we set up thresholds for each dimension on the basis of the coordinates of the midpoint. In this way, we could balance training data in each category.

For AllMusic dataset, the original annotation is quadrants. In accordance with our predictive targets and the annotation used in PMEmo dataset, we transform quadrants into binary valence and arousal values.

4.4 Training Model Setup

In the unsupervised learning stage, the masked data is fed into a CNN-based autoencoder model. The parameters of the proposed neural networks are given in Table 2. All of the 2D CNN layers specify 3×3 kernel size with one stride. 2×2 pool size with stride length of 2 is applied for 2D maxpooling layers and same size is applied for 2D upsampling layers as well. The filter size starts with 128 and decreases layer by layer in the encoder, then increases correspondingly in the decoder ending with 1 to return to the initial shape. During optimization, the L2 regularizer applies a penalty to the output of the first CNN layer with a 0.001 learning rate to benefit model convergence. Once the training is finished, we save the optimal weights of the encoder module.

In the supervised learning stage, we assemble temporal segment-level representations in sequence through the saved encoder module, and then put them into the BiLSTM model. We set the output units of the LSTM layers as 512 for forward and backward direction separately. After that, the dropout rate of 0.5 is applied. The final binary classification is obtained through the dense layer with the softmax activation. In this part, we also consider LSTM and GRU (Gated Recurrent Unit) models instead due to less parameters and training cost. However, BiLSTM model could capture sequential information in both directions and has higher performance in the experiment. Then we check the detail of training cost for BiLSTM model: the training time for each epoch is generally 5s–21s and the number of epochs for each fold is averagely 25. Based on this, time cost is completely affordable. Therefore, we give priority to performance and choose BiLSTM model.

We evaluate the whole model by running 10-fold cross validation and obtaining the average performance based on classification accuracy and F1-score. Accordingly, we split training/test sets with the ratio of 9:1. In each fold, we run 5 rounds for Valence/Arousal predictions respectively to check the statistical results. For both unsupervised learning and supervised learning, the Adam optimizer [42] is used, and the early stopping strategy is configured with the patience of 10-epoch for the validation dataset to avoid overfitting during training. The former model monitors reconstruction Huber loss while the latter model monitors classification accuracy. The details of some hyperparameters are summarized in Table 3. Moreover,

Table 2 The parameters of the proposed autoencoder model

Layer Type	Parameters	Output Shape
Input	-	(216, 128, 1)
2D CNN	kernel=3 × 3, stride=1, filter=128	(216, 128, 128)
2D Maxpooling	pool_size=2 × 2, stride=2	(108, 64, 128)
2D CNN	kernel=3 × 3, stride=1, filter=64	(108, 64, 64)
2D Maxpooling	pool_size=2 × 2, stride=2	(54, 32, 64)
2D CNN	kernel=3 × 3, stride=1, filter=32	(54, 32, 32)
2D Maxpooling	pool_size=2 × 2, stride=2	(27, 16, 32)
2D CNN	kernel=3 × 3, stride=1, filter=32	(27, 16, 32)
2D Upsampling	size=2 × 2	(54, 32, 32)
2D CNN	kernel=3 × 3, stride=1, filter=64	(54, 32, 64)
2D Upsampling	size=2 × 2	(108, 64, 64)
2D CNN	kernel=3 × 3, stride=1, filter=128	(108, 64, 128)
2D Upsampling	size=2 × 2	(216, 128, 128)
2D CNN	kernel=3 × 3, stride=1, filter=1	(216, 128, 1)

Table 3 The hyper-parameters for model training

Hyper-parameter	Unsupervised Learning	Supervised Learning
Optimizer	Adam	Adam
Optimizer's Learning Rate	1e-3	1e-5
Batch Size	64	10
Loss	Huber	Categorical Cross Entropy

we report the general time cost of our model training on two datasets (see Table 4). All experiments are implemented via Nvidia GeForce GTX 1080 GPU. The unsupervised learning usually takes 100–200 epochs per fold. The supervised learning usually takes 20-30 epochs per fold.

To validate the advantage of the proposed autoencoder model, we also build up a baseline model that combines CNN and BiLSTM directly. The CNN module reuses the structure of the feature encoder in the unsupervised learning, followed by BiLSTM for emotion classification. These two parts are trained together.

5 Results

In this section, we report our experiment results based on selected segment duration and compare our performance with previous work.

5.1 Performance of Different Segment Duration

The segments of different duration have been applied in our experiments. In multiple runs for each segment duration, we average 10-fold scores. The results are shown in Table 5, and show that the performance for arousal recognition is always better than valence in all of the segment lengths investigated. The results also indicate that shorter segment length shows better performance on the valence dimension while longer segment duration benefits arousal performance. For example, in PMEmo dataset, 1-second segment shows the best valence results with 79.01% accuracy and 83.2% F1-score while 5s/10s's segments show better accuracy (83.62%/83.51%) and F1-score (86.52%/86.62%) on arousal dimension. AllMusic dataset shows the similar trends. For such results, we analyze the possible reasons in the discussion section.

5.2 Performance Comparison with Different Models and Sources

Table 6 shows a performance comparison with cutting-edge benchmarks based on different models and sources. From this comparison it is clear that our model can outperform any models using a single data source, either music or electrodermal activity signals. Compared to the Yin et al.

Table 4 The general time cost of the proposed model during training

Dataset	Segment Duration	Unsupervised Learning (CNN-based autoencoder)	Supervised Learning (BiLSTM)
PMEmo	1s	75s/epoch, 3h/fold	21s/epoch, 525s/fold
	3s	24s/epoch, 1h/fold	13s/epoch, 325s/fold
	5s	15s/epoch, 0.6h/fold	7s/epoch, 175s/fold
	10s	8s/epoch, 0.3h/fold	5s/epoch, 138s/fold
AllMusic	3s	28s/epoch, 1.1h/fold	18s/epoch, 450s/fold
	5s	18s/epoch, 0.7h/fold	13s/epoch, 325s/fold
	10s	9s/epoch, 0.4h/fold	9s/epoch, 225s/fold

In this table, 's' denotes second, 'h' denotes hour.

Table 5 The performance comparison based on different segment duration

Dataset	Segment Duration	Valence		Arousal	
		Accuracy	***F1-score***	***Accuracy***	***F1-score***
PMEmo	1s	**79.01%**	**83.2%**	83.19%	86.1%
	3s	78.75%	82.95%	82.67%	85.59%
	5s	78.23%	82.64%	**83.62%**	**86.52%**
	10s	77.58%	82.18%	**83.51%**	**86.62%**
AllMusic	3s	**67.11%**	**67.11%**	85.67%	85.67%
	5s	66.89%	66.89%	**86.56%**	**86.56%**
	10s	66.45%	66.45%	86.11%	86.11%

model [43] that uses music sources only, the accuracy for valence prediction in our model increases by more than 12% and the corresponding F1-score increases by more than 10%. Similarly, there are increases of almost 17% and 13% on arousal recognition in terms of accuracy and F1-score respectively. Our model even competes with the latest multimodal framework [18] that utilizes EDA signals and music together with attention neural networks. Furthermore, we compared our model with the baseline model which also use segment-level inputs but lack of the autoencoder architecture. The results show that our model is superior to the baseline model in both emotion dimensions.

6 Discussion

6.1 Segment Duration Analysis

From Table 5, we may suppose that a longer segment length contains more acoustic cues for arousal recognition while a shorter one has more relevant information for valence prediction. Compared with segments of long duration, shorter segments are more likely to avoid changes of musical characteristics and reflect consistent perceptual properties of music like harmony, pitches that benefit valence recognition [45]. In contrast, relatively long duration may capture more time-domain regularities like beat and tempo that benefit arousal recognition [46]. Further, we conduct paired t-tests to examine the performance of different segment duration, the results demonstrate that there is no statistical significance with respect to which segment duration is best. The possible reason is that we use log-mel spectrogram with same input

Table 6 The performance comparison with different models and different sources based on PMEmo dataset

Models	Core Methods	Input Source	Audio Segmentation	Valence		Arousal	
				Accuracy	*F1-score*	*Accuracy*	*F1-score*
RTCAN-1D [18]	attention module + ResNet + openSMILE	EDA + Music	No	77.30%	80.94%	82.51%	85.62%
RTCAG [18]	attention module + ResNet	EDA	-	63.61%	62.47%	64.05%	64.82%
SVM[43]	SVM	Music	No	70.43%	75.32%	71.49%	76.36%
SVM[44]	SVM	Music + Lyrics	No	61.98%	-	68.75%	-
The baseline	CNN + BiLSTM	Music	Yes	77.44%	81.91%	82.79%	85.17%
Our model	CNN-based autoencoder + BiLSTM	Music	Yes	**79.01%**	**83.2%**	**83.62%**	**86.52%**

shape for different segment duration, which limits the selection of the length of FFT window and the hop length thereby impacting the musical pattern extraction from audio data. Another reason is to what extent the segment duration could match with the emotional boundary. The performance depends on whether the fixed segmentation could cover emotional segmentation well for most of songs [33]. Generally, 5-second segment is a relatively better choice for our model as this duration is a reasonable trade-off between performance and computing cost.

6.2 Performance Analysis Compared with Other Models

In this part, we discuss segment-based framework and model structures. Compared with the models in Table 6, our model using segment-level learning shows better performance than other models that used the whole music excerpts directly. The long duration may contain acoustic cue variations and emotional state changes [4], which may make learning models confused and have difficulty extracting unified musical features targeted to one kind of emotion [33]. Segment-based learning relieves this problem as the relatively shorter duration usually reflects consistent music feature patterns that facilitate emotion recognition and improve the effectiveness of learning [32]. On the other hand, we compared two models with audio segmentation. Under the same experimental circumstance, our model with the autoencoder structure outperforms the baseline model. It is demonstrated that the autoencoder can contribute to the increase of final performance. The advantage is that the autoencoder makes it possible to separate two-stage training with their own optimum parameters. In the meantime, as an unsupervised learning method, no labels are required. Further, our segment-level unsupervised learning brings about more flexibility of model structure design. The framework is divided into 2 parts, one part concentrates on feature representation while the other part focuses on target prediction. It is possible that we could replace one part without changing the other part as long as the data interfaces could match well with each other meaningfully. For example, another effective deep neural network is used to predict final emotion instead of the LSTM model. This approach could be considered in future research.

Another factor we consider is the cost. The state-of-the-art work adopted attention mechanisms [18]. This is powerful for learning music representations, but it introduces more training parameters and increases the complexity of computing which requires more computing resources and aggravates the burden of operating environment, even more time cost [47]. We replaces the attention architecture with stacked convolutional neural networks, which reduces the time cost (refer to Table 4) but achieves the equivalent results. We argue that our model is generally more cost-effective.

6.3 Ablation Test for Masking Data

We carry out the ablation test to examine the effectiveness of the masking methods. The 10 folds of accuracy and F1-score for valence/arousal recognition are visualized in Fig. 4. In each subfigure, both lines represent the performance of the model without masking and the model with masking. For PMEmo dataset, both lines go across each

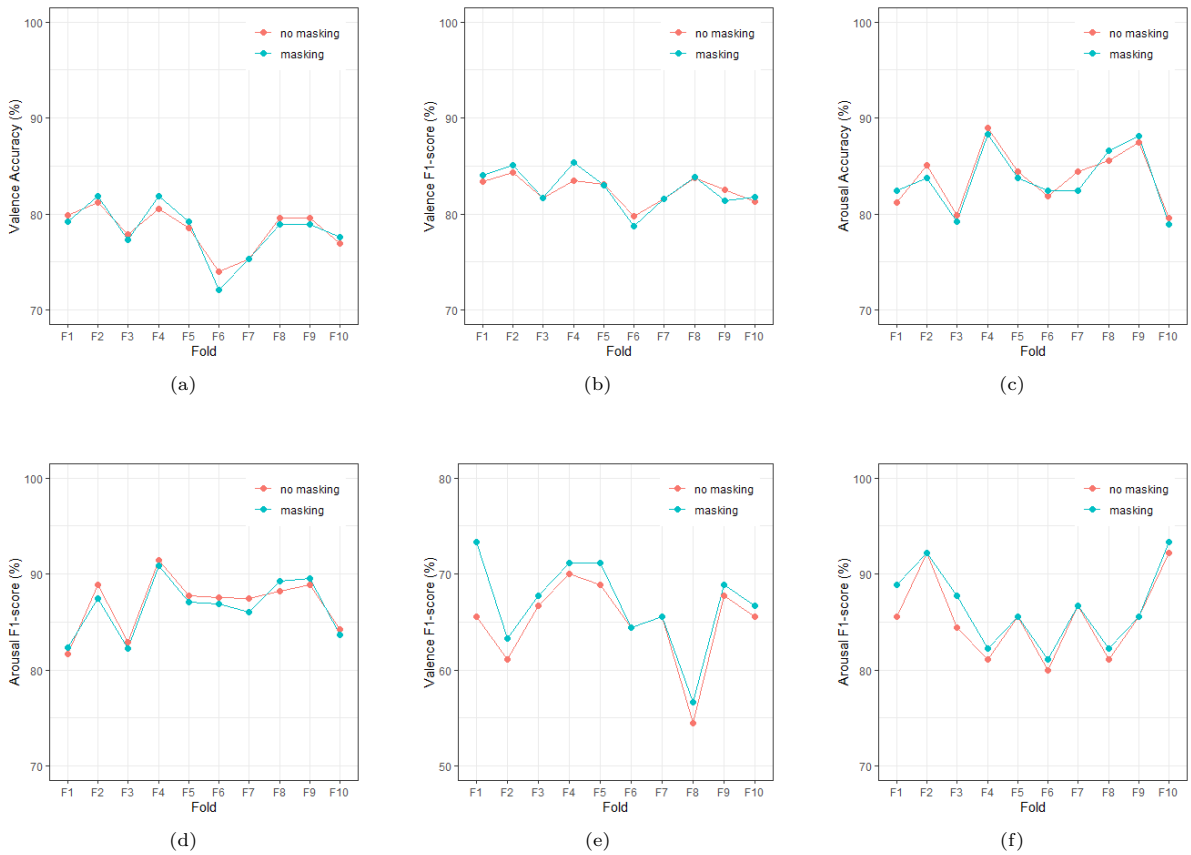


Fig. 4 Masking impact on the performance. For PMEmo dataset: (a) shows the accuracy for valence; (b) shows F1-score for valence; (c) shows the accuracy for arousal; (d) shows F1-score for arousal. For AllMusic dataset: (e) shows F1-score for valence; (f) shows F1-score for arousal; the accuracy comparison is same as F1-score.

other several times. This result can be explained by characteristics of the dataset. As the chorus part of a popular song contains the repetition of musical content that shows more clear and intense emotion expression [48]. Such data morphology decreases the data variation and the outliers so as to lessen the effect of masking methodology. For AllMusic dataset, it contains different genres of songs and balanced training samples. The effectiveness of masking is statistically significant. Overall, we think that masking could benefit the model robustness. In the future work, we may investigate the effect of different proportions of masking span on performance.

7 Conclusion

In this paper, we propose a segment-level two-stage learning framework. This naturally combines the unsupervised learning as a feature extractor with the supervised learning as a music emotion classifier. First, we use a CNN-based autoencoder to calculate feature representations for contiguous segments that make up each music excerpt. And then, the time-series segments are fed into the BiLSTM model to predict emotion for this music excerpt. In this way, we implement segment-level feature extraction without being limited to song-level annotation. Additionally, we apply the time/frequency masking approach to the segment inputs for enhancing model robustness. The experimental results show that our model achieves better performance than those models using a single feature source, even competing with the cutting-edge multi-modal framework. Compared with the whole music excerpts as model inputs, segments with relatively short duration increase

the data scale and contain less change of acoustic cues. Due to this, the learning models could detect the correlation between musical features and emotion more effectively. Apart from that, this two-stage training framework is more flexible and makes changing the combinations of neural networks possible. That means much potential for performance improvement.

References

[1] Jeon B, Kim C, Kim A, et al (2017) Music emotion recognition via end-To-end multimodal neural networks. In: CEUR Workshop Proceedings

[2] HE N, Ferguson S (2021) Multi-view Neural Networks for Raw Audio-based Music Emotion Recognition. In: 2020 IEEE International Symposium on Multimedia (ISM). IEEE, pp 168–172, https://doi.org/10.1109/ism.2020.00037

[3] Aljanaki A, Yang YH, Soleymani M (2017) Developing a benchmark for emotional analysis of music. PLoS ONE 12(3). https://doi.org/10.1371/journal.pone.0173392

[4] Xiao Z, Dellandrea E, Dou W, et al (2008) What is the best segment duration for music mood analysis ? In: 2008 International Workshop on Content-Based Multimedia Indexing, CBMI 2008, Conference Proceedings, pp 17–24, https://doi.org/10.1109/CBMI.2008.4564922

[5] Yang YH, Chen HH (2012) Machine Recognition of Music Emotion: A Review. ACM Transactions on Intelligent Systems and Technology 3(3):1–30. https://doi.org/10.1145/2168752.2168754

[6] Sarkar R, Choudhury S, Dutta S, et al (2020) Recognition of emotion in music based on deep convolutional neural network. Multimedia Tools and Applications 79(1-2):765–783. https://doi.org/10.1007/s11042-019-08192-x

[7] Park DS, Chan W, Zhang Y, et al (2019) Specaugment: A simple data augmentation method for automatic speech recognition. In: Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, pp 2613–2617, https://doi.org/10.21437/Interspeech.2019-2680

[8] Lee J, Park J, Kim KL, et al (2018) SampleCNN: End-to-end deep convolutional neural networks using very small filters for music classification. Applied Sciences (Switzerland) 8(1). https://doi.org/10.3390/APP8010150

[9] Choi K, Fazekas G, Sandler M (2016) Automatic tagging using deep convolutional neural networks. In: Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016, pp 805–811

[10] Wang Q, Su F, Wang Y (2019) A hierarchical attentive deep neural network model for semantic music annotation integrating multiple music representations. In: ICMR 2019 - Proceedings of the 2019 ACM International Conference on Multimedia Retrieval. Association for Computing Machinery, Inc, pp 150–158, https://doi.org/10.1145/3323873.3325031

[11] Défossez A, Usunier N, Bottou L, et al (2019) Music Source Separation in the Waveform Domain. arXiv URL http://arxiv.org/abs/1911.13254

[12] de Berardinis J, Cangelosi A, Coutinho E (2020) The multiple voices of musical emotions: source separation for improving music emotion recognition models and their interpretability. In: Proceedings of the 21st International Society for Music Information Retrieval Conference, pp 310–217, URL https://www.ismir2020.net/assets/img/proceedings/2020_ISMIR_Proceedings.pdf

[13] Li J, Gao S, Han N, et al (2015) Music Mood Classification via Deep Belief Network. In: 2015 IEEE International Conference on Data Mining Workshop (ICDMW), pp 1241–1245, https://doi.org/10.1109/ICDMW.2015.136

[14] Corona H, O'Mahony MP (2015) An exploration of mood classification in the million songs dataset. In: Proceedings of the 12th International Conference in Sound and Music

Computing, SMC 2015, pp 363–370

[15] Tripathi S, Acharya S, Sharma R, et al (2017) Using Deep and Convolutional Neural Networks for Accurate Emotion Classification on DEAP Dataset. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, pp 4746–4752, URL https://www.aaai.org/ocs/index.php/IAAI/IAAI17/paper/view/15007/13731

[16] Kadambari KV, Bhattacharya A (2018) A Multimodal approach towards emotion recognition of music using audio and lyrical content. arXiv URL http://arxiv.org/abs/1811.05760

[17] Delbouys R, Hennequin R, Piccoli F, et al (2018) Music mood detection based on audio and lyrics with deep neural net. In: Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, pp 370–375

[18] Yin G, Sun S, Yu D, et al (2020) A Efficient Multimodal Framework for Large Scale Emotion Recognition by Fusing Music and Electrodermal Activity Signals. URL http://arxiv.org/abs/2008.09743

[19] Hu X, Choi K, Downie JS (2017) A framework for evaluating multimodal music mood classification. Journal of the Association for Information Science and Technology 68(2):273–285. https://doi.org/10.1002/asi.23649

[20] Lian Z, Li Y, Tao J, et al (2018) Investigation of Multimodal Features, Classifiers and Fusion Methods for Emotion Recognition. URL http://arxiv.org/abs/1809.06225

[21] Zhou J, Chen X, Yang D (2019) Multimodel music emotion recognition using unsupervised deep neural networks. In: Lecture Notes in Electrical Engineering, https://doi.org/10.1007/978-981-13-8707-4{_}3

[22] Xianyu H, Xu M, Wu Z, et al (2016) Heterogeneity-entropy based unsupervised feature learning for personality prediction with cross-media data. In: Proceedings - IEEE International Conference on Multimedia and Expo, https://doi.org/10.1109/ICME.2016.7552980

[23] Fan J, Yang YH, Dong K, et al (2020) A Comparative Study of Western and Chinese Classical Music Based on Soundscape Models. In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pp 521–525, https://doi.org/10.1109/ICASSP40776.2020.9052994

[24] Hershey S, Chaudhuri S, Ellis DP, et al (2017) CNN architectures for large-scale audio classification. In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pp 131–135, https://doi.org/10.1109/ICASSP.2017.7952132

[25] Zhao Y, Wu X, Ye Y, et al (2020) MusiCoder: A Universal Music-Acoustic Encoder Based on Transformers. https://doi.org/10.1007/978-3-030-67832-6{_}34

[26] Senac C, Pellegrini T, Mouret F, et al (2017) Music feature maps with convolutional neural networks for music genre classification. In: ACM International Conference Proceeding Series, vol Part F1301. ACM, p 19, https://doi.org/10.1145/3095713.3095733

[27] Choi K, Fazekas G, Sandler M, et al (2017) Convolutional recurrent neural networks for music classification. In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pp 2392–2396, https://doi.org/10.1109/ICASSP.2017.7952585

[28] Madiraju NS, Sadat SM, Fisher D, et al (2018) Deep Temporal Clustering : Fully Unsupervised Learning of Time-Domain Features. URL http://arxiv.org/abs/1802.01059

[29] Devlin J, Chang MW, Lee K, et al (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, pp 4171–4186

[30] Liu AT, Yang SW, Chi PH, et al (2020) Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders. In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pp 6419–6423, https://doi.org/10.1109/ICASSP40776.2020.9054458

[31] Fu C, Liu C, Ishi CT, et al (2020) Multi-modality emotion recognition model with gat-based multi-head inter-modality attention. Sensors (Switzerland) 20(17):1–15. https://doi.org/10.3390/s20174894

[32] Wu B, Zhong E, Horner A, et al (2014) Music emotion recognition by multi-label multi-layer multi-instance multi-view learning. In: MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia, pp 117–126, https://doi.org/10.1145/2647868.2654904

[33] Aljanaki A, Wiering F, Veltkamp RC (2015) Emotion based segmentation of musical audio. In: Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015, pp 770–776

[34] Russell JA (1980) A circumplex model of affect. Journal of Personality and Social Psychology 39(6):1161–1178. https://doi.org/10.1037/h0077714

[35] Girshick R (2015) Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1440–1448, https://doi.org/10.1109/ICCV.2015.169

[36] Zhang K, Zhang H, Li S, et al (2018) The PMEmo dataset for music emotion recognition. In: ICMR 2018 - Proceedings of the 2018 ACM International Conference on Multimedia Retrieval, pp 135–142, https://doi.org/10.1145/3206025.3206037

[37] Panda R, Malheiro R, Paiva RP (2018) Novel Audio Features for Music Emotion Recognition. IEEE Transactions on Affective Computing 11(4):614–626. https://doi.org/10.1109/TAFFC.2018.2820691

[38] Warriner AB, Kuperman V, Brysbaert M (2013) Norms of valence, arousal, and dominance for 13,915 English lemmas. Behavior Research Methods 45(4):1191–1207. https://doi.org/10.3758/s13428-012-0314-x

[39] Bigand E, Vieillard S, Madurell F, et al (2005) Multidimensional scaling of emotional responses to music: The effect of musical expertise and of the duration of the excerpts. Cognition and Emotion 19(8):1113–1139. https://doi.org/10.1080/02699930500204250

[40] Nordström H, Laukka P (2019) The time course of emotion recognition in speech and music. The Journal of the Acoustical Society of America 145(5):3058–3074. https://doi.org/10.1121/1.5108601

[41] Yin Z, Wang Y, Liu L, et al (2017) Cross-subject EEG feature selection for emotion recognition using transfer recursive feature elimination. Frontiers in Neurorobotics 11(APR). https://doi.org/10.3389/fnbot.2017.00019

[42] Kingma DP, Ba JL (2015) Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings

[43] Yin G, Sun S, Zhang H, et al (2019) User Independent Emotion Recognition with Residual Signal-Image Network. In: Proceedings - International Conference on Image Processing, ICIP, pp 3277–3281, https://doi.org/10.1109/ICIP.2019.8803627

[44] Sharma H, Gupta S, Sharma Y, et al (2020) A New Model for Emotion Prediction in Music. In: 2020 6th International Conference on Signal Processing and Communication, ICSC 2020, pp 156–161, https://doi.org/10.1109/ICSC48311.2020.9182745

[45] Gabrielsson A, Lindström E (2001) The influence of musical structure on emotional expression. In: Music and emotion: Theory and research. p 223–248

[46] Grekow J (2017) Audio features dedicated to the detection of arousal and valence in music recordings. In: Proceedings - 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications, INISTA 2017, pp 40–44, https://doi.org/10.1109/INISTA.2017.8001129

[47] He K, Sun J (2015) Convolutional neural networks at constrained time cost. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 5353–5360, https://doi.org/10.1109/CVPR.2015.7299173

[48] Yeh CH, Tseng WY, Chen CY, et al (2014) Popular music representation: chorus detection & emotion recognition. Multimedia Tools and Applications 73(3):2103–2128. https://doi.org/10.1007/s11042-013-1687-2