國立臺灣大學電機資訊學院電信工程學研究所

碩士論文

Graduate Institute of Communication Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

基於對抗式適應之跨文化音樂情緒辨識

Cross-Cultural Music Emotion Recognition by Adversarial Discriminative Domain Adaptation

陳怡瑋

Yi-Wei Chen

指導教授：陳宏銘 博士

Advisor: Homer H. Chen, Ph.D.

中華民國 107 年 8 月

August, 2018

doi:10.6342/NTU201802248

國立臺灣大學碩士學位論文
口試委員會審定書

基於對抗式適應之跨文化音樂情緒辨識
Cross-Cultural Music Emotion Recognition by Adversarial Discriminative Domain Adaptation

本論文係陳怡瑋君（R02942096）在國立臺灣大學電信工程學研究所完成之碩士學位論文，於民國 107 年 7 月 12 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

陳宏銘　（簽名）

（指導教授）

楊奕軒　　王新民

蘇黎　　蔡銘峰

所　　長　吳宗霖　（簽名）

誌謝

論文能夠順利完成，首先，非常感恩我的指導教授 陳宏銘老師，一步一腳印的陪伴教導，老師不怕教你只怕你不願意學，剛開始的研究態度不是很認真，但從老師身上看見自己的不足，體會到成功是需要一步一腳印去實踐的!慢慢才比較上軌道，老師雖然嚴格但對研究要求的態度令我相當佩服且收穫良多。再來要非常感恩 楊奕軒博士，從碩士班第三年起就給予我在研究上以及心理上相當多的幫忙，每次快做不下去時，學長總是會適時給予許多的想法以及給予心理建設，從學長身上也看到了無私，若沒有學長的幫忙是沒辦法完成這篇論文的，非常感恩。

再來要感恩實驗室學長鍾佳豪(Mike)，從剛開始研究卡關，Mike 就一直給予許多的協助，一路陪伴協助真的感恩在心，祝你博班順利畢業。另外要感恩的是實驗室學弟林奕勳，我們就像戰友一樣一起面對一樣的困難，一起突破。再來要感謝音樂組的每位學弟學妹們，因為有你們給予我許多有幫助的建議，我才得以順利完成論文。

最後，要感恩我的父母以及家人，感恩你們總是默默為我打理一切，讓我可以無憂的重回研究所，抱歉，讓你們擔心了！感恩議芳，壓力大的時候會不小心語氣不好傷害了你，但你卻還是包容與陪伴我。感恩冠廷，每次你真心的鼓勵讓我可以找回失去的信心。感恩所有在我面對研究窘境時，給予我幫忙與鼓勵的每個人。感恩我的 師父 妙禪師父，感恩 師父安住了我的心，讓我遭遇困難的時候選擇面對不逃避，讓我明白自己的傲慢與懶散，更體會未來不論有多少挑戰，有 師父的護祐，能夠充滿信心不擔心地去面對，誓願追隨 師父成佛利眾。

doi:10.6342/NTU201802248

中文摘要

對於建立自動音樂情緒辨識系統而言，收集音樂給人的情緒感受標記是必須的。迄今，大部分的音樂情緒資料集都是以西洋歌曲為主。若音樂情緒辨識系統是以西洋曲風的資料集建立的，此系統可能沒辦法適用於非西洋曲風的歌曲，因這兩個曲風受文化背景的影響，在音樂特徵上以及標記者的情緒感受上皆有不同之處。即使這樣的問題已在跨文化以及跨資料集的研究中被發現，但很少有研究探討如何將用收集到的曲風資料集訓練的模型重新訓練以適應於我們感興趣的曲風上。在本篇論文中，我們提出以非監督式對抗式域適應之方法來解決這個問題。此方法應用了類神經網路之模型使兩曲風學到的表徵無法被區分。又情緒感受本身包含了許多面向，因此我們考慮了與音色、音高、以及節奏性相關之三種輸入特徵來評估模型之成效。結果顯示以西洋流行歌曲訓練的模型透過我們提出的方法可大幅改善用中文歌曲預測情緒正負向的準確率。

關鍵字：跨文化、音樂情緒辨識、音樂資訊檢索、域適應、對抗式判別之域適應

ABSTRACT

Annotation of the perceived emotion of a music piece is needed for an automatic music emotion recognition system. To date, the majority of music emotion datasets are for Western pop songs. A music emotion recognizer trained on such datasets may not work well for non-Western pop songs due to the differences in acoustic characteristics and emotion perception that are inherent to cultural background. Although the problem was also found in cross-cultural and cross-dataset studies, little has been done to learn how to adapt a model pre-trained on a *source* music genre to a *target* music genre of interest. In this paper, we propose to address the problem by an unsupervised adversarial domain adaptation method. It employs neural network models to make the target music indistinguishable from the source music in a learned feature representation space. Because emotion perception is multifaceted, three types of input features related to timbre, pitch, and rhythm are considered for performance evaluation. The results show that the proposed method effectively improves the prediction of the valence of Chinese pop songs from a model trained for Western pop songs.

Keywords – Cross-cultural, music emotion recognition, music information retrieval, domain adaptation, adversarial discriminative domain adaptation.

doi:10.6342/NTU201802248

CONTENTS

doi:10.6342/NTU201802248

LIST OF FIGURES

doi:10.6342/NTU201802248

LIST OF TABLES

Chapter 1 INTRODUCTION

With the popularity of on-line music services, a large amount of music pieces created from different corners of the world can be accessed by global audiences. Automatic music emotion recognition (MER) techniques have been developed [1], [2] to facilitate such global music information retrieval [3] by exploiting the appealing feature of music listening that music evokes human mood or emotion.

However, existing MER datasets are mostly created for Western pop songs [3], [4]. Using an MER model trained on such songs to predict the emotion of non-Western music pieces does not yield the best performance [5], [6] due to the cultural differences in acoustic characteristics and emotion perception. Although some efforts have been made to enrich non-Western datasets, the size of such datasets is much smaller than that of Western datasets [4]. To deal with the deficiency, cross-dataset transferring or generalization seems a feasible approach.

Hu and Yang [7] studied the cross-cultural and cross-dataset generalizability of MER models trained on different music features for predicting valence and arousal values, the two principal dimensions of emotions [1]. They found that arousal can be better predicted across datasets than valence.

doi:10.6342/NTU201802248

The issue that a machine learning model pre-trained on a source dataset may not perform well for a target dataset is not specific to MER. It is known as a *domain shift* issue resulting from the data distribution biases between the source dataset and the target dataset [8], [9]. There are two typical machine learning solutions. The first solution is to fine tune the model using the target dataset [10]. However, as the size of the target dataset is usually small, the model may easily overfit. The second solution, referred to as *domain adaptation*, is to learn a cross-domain invariant feature representation by minimizing the discrepancy between the target data and source data [14]–[18]. It is usually achieved by unsupervised learning. Neither solution has been employed in prior work for MER, to our best knowledge.

In this paper, we study whether and how an unsupervised adversarial domain adaptation method can improve cross-cultural MER. Moreover, as the perception of music emotion is multifaceted, three types of acoustic features related to timbre, pitch, and rhythm are considered for the investigation of cross-cultural generalizability. We conduct experiments on AMG1608 (a Western pop music dataset) and CH818 (a Chinese pop music dataset) used in a similar study [7].

In what follows, we first discuss related work on MER and domain adaptation. Then, we describe the proposed adversarial discriminative domain adaptation method, our

doi:10.6342/NTU201802248

network architecture, and experimental settings. Finally, we discuss the experimental results and draw some concluding remarks.

doi:10.6342/NTU201802248

Chapter 2 RELATED WORK

2.1 Music Emotion Recognition

Music emotion is often represented using either the categorical model or the dimensional model developed in music psychology. The categorical model uses a set of discrete mood labels, such as sad and happy, to describe music emotion. Each song is assigned at least one label. The dimensional model represents music emotion in a low-dimensional space, such as valence and arousal [1], by continuous values. The class of MER techniques for predicting the emotion categories of music pieces is referred to as music emotion classification, and the class of MER techniques for predicting the numerical emotion values of music pieces is referred to as music emotion regression. Both classes of techniques have been adopted in many studies, using music of the same genre or cultural background for training and testing [1]–[3].

While previous studies focused on training MER models with Western music datasets, some recent work started to investigate whether such models can be directly applied to non-Western music [7], [11]–[13]. For example, Hu and Yang [12] explored six music related features for music emotion classification of English and Chinese songs and found that arousal prediction works generally well across datasets, but valence prediction is culture-dependent. Similarly, the study reported by Eerola [13] shows that arousal

doi:10.6342/NTU201802248

prediction is generalizable across different musical genres, whereas valence prediction is not.

As acoustic features account for different music characteristics, Hu and Yang [7] further investigated the generalizability of different feature sets for music emotion regression. They found that features related to loudness and timbre have better generalizability for both valence and arousal, but rhythm-related features are only effective for valence and pitch-related features are only effective for arousal. However, transferring useful information from Western music to non-Western music is not explored in these studies.

2.2 Domain Adaptation

Recent domain adaptation methods can be categorized into two approaches. The first approach aims to reweight a model pre-trained on the source domain to make the learned feature representation general enough for the target domain as well. The learning is performed by minimizing a *domain distance metric*, such as maximum mean discrepancy [14], [15] or correlation distance [16], [17]. Alternatively, one can also simultaneously train a common representation for classification and reconstruction [18].

Instead of using pre-defined distance metrics, the second approach trains a model to measure the discrepancy between source and target domains, in a data-driven way. Adversarial adaptation methods belong to this approach and have gained popularity

doi:10.6342/NTU201802248

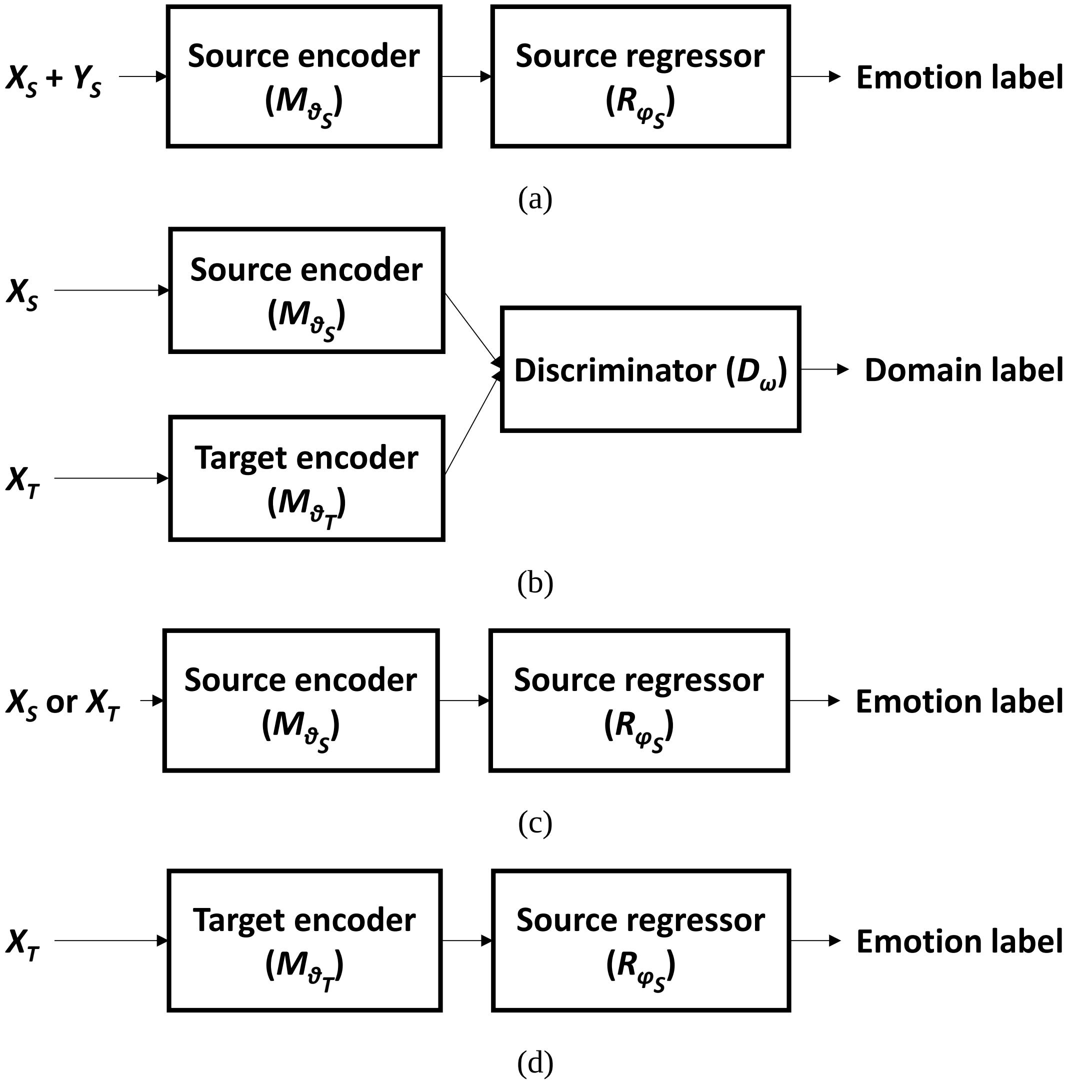
recently, after the success of generative adversarial network (GAN) [19]. The goal of a GAN is to estimate a *generative model* via an adversarial process that simultaneously trains two models: a generative model *G* that tries to generate artificial data with distribution similar to the training data, and a *discriminative model D* that aims to distinguish (through binary classification) between the real data and the data created by *G*. The process is adversarial, because the objective of *D* is to maximize the classification accuracy, whereas the objective of *G* is to minimize the classification accuracy. *D* and *G* are trained iteratively, in a hope that by the end of the process the output of *G* looks similar to the real data. In the same vein, adversarial domain adaptation aims to train a generative model *G* that transforms data from the target domain in such a way that makes *D*, which is a domain classifier, believe that the output of *G* are data from the source domain.

Among various adversarial adaptation methods, we choose the adversarial discriminative domain adaptation (ADDA) method [8] in this work, because it has been proven successful for various transfer learning tasks. There are several extensions [20]–[22], but ADDA is one of the earliest methods of its kind and does not require a generative model.

doi:10.6342/NTU201802248

Chapter 3 METHODOLOGY

In this section, we discuss how to apply ADDA to cross-cultural MER. We assume that we are given the source data X_S (i.e. input audio features), the source labels Y_S (i.e. emotion labels), and the target data X_T, but not the target labels Y_T. The training process



$X_S + Y_S$
Source encoder (M_{ϑ_S})
Source regressor (R_{φ_S})
Emotion label
(a)
X_S
X_T
Source encoder (M_{ϑ_S})
Target encoder (M_{ϑ_T})
Discriminator (D_ω)
Domain label
(b)
X_S or X_T
Source encoder (M_{ϑ_S})
Source regressor (R_{φ_S})
Emotion label
(c)
X_T
Target encoder (M_{ϑ_T})
Source regressor (R_{φ_S})
Emotion label
(d)

Figure 3.1. System flow, where X_S are the source data, Y_S are the source emotion labels, and X_T are the target data. (a) The system flow of pre-training an MER model. (b) The system flow of adversarial discriminative domain adaptation. (c) The system flow of testing a dataset without adaptation. (d) The system flow of testing a dataset with adaptation. and the testing system flow of the model without and with adaptation.

doi:10.6342/NTU201802248

of the proposed method is illustrated in Figs. 3.1(a) and 3.1(b) and the test process in Fig. 3.1(d).

3.1 Pre-training

As shown in Fig. 3.1(a), the training process starts with a pre-training phase, using data from the source domain only. Given source data X_S and source labels Y_S, we train a deep neural network for emotion prediction. The task of the first few layers (which can be convolutional layers) of the deep neural network is to perform feature extraction. It projects the input feature representation X_S into a learned feature space. The task of the last few layers (which can be fully connected layers) is to predict the emotion values based on the learned features. Therefore, we call the first few layers as a *source encoder*, and denote it by M_{θ_S}, and the last few layers as a *source regressor* and denote it by R_{φ_S}. The source encoder and the source regressor are trained jointly by minimizing the mean squared error L_r between Y_S and the predicted emotion values,

$$L_r = \frac{1}{N}\sum_{n=1}^{N}\left(y_S{}^{(n)} - R_{\varphi_S}\left(M_{\theta_S}\left(x_S{}^{(n)}\right)\right)\right)^2, \tag{1}$$

where N denotes the batch size of source data and $\left\{x_S{}^{(n)}, y_S{}^{(n)}\right\}_{n=1}^{N}$ denotes a batch of $\{X_S, Y_S\}$. A pseudo code of the pre-training MER model is described in Table 3.1.

Table 3.1. The pseudo code of the pre-training MER algorithm

```
Require： $\alpha$, the learning rate. $N$, the batch size.
Require：$\theta_{S_0}$, weights of the initial source encoder. $\varphi_{S_0}$, weights of the initial source regressor.
while $\theta_S$ and $\varphi_S$ have not converged do
    Sample $\{x_S{}^{(n)}, y_S{}^{(n)}\}_{n=1}^{N}$ a batch of the $\{X_S, Y_S\}$.
    $g_{\theta_S} \leftarrow \nabla_{\theta_S} [\frac{1}{N}\sum_{n=1}^{N}\left(y_S{}^{(n)} - R_{\varphi_S}\left(M_{\theta_S}\left(x_S{}^{(n)}\right)\right)\right)^2]$
    $g_{\varphi_S} \leftarrow \nabla_{\varphi_S} [\frac{1}{N}\sum_{n=1}^{N}\left(y_S{}^{(n)} - R_{\varphi_S}\left(M_{\theta_S}\left(x_S{}^{(n)}\right)\right)\right)^2]$
    $\theta_S \leftarrow \theta_S - \alpha \cdot \mathrm{Adam}(\theta_S;\ g_{\theta_S})$
    $\varphi_S \leftarrow \varphi_S - \alpha \cdot \mathrm{Adam}(\varphi_S;\ g_{\varphi_S})$
end while
```

As shown in Fig. 3.1(c), we can feed target data X_T as the input to the source encoder and the source regressor for emotion prediction. But, due to the domain shift issue, the source regressor may not perform well for the target data.

3.2 Adversarial Discriminative Domain Adaptation

As shown in Fig. 3.1(d), ADDA attempts to address the domain shift issue by learning a *target encoder* M_{θ_T} for the target data. It is assumed that the output $M_{\theta_T}(X_T)$ of the target encoder for the target data would have similar distribution as the output $M_{\theta_S}(X_S)$ of the source encoder. If this is achieved, we consider that the domain shift issue is mitigated and that we can use the source regressor to predict the emotion for the target data without the need of training a target regressor.

The key of ADDA is to learn the target encoder. This is achieved by using a *discriminator* D_ω, which takes either the source feature representation $M_{\theta_S}(X_S)$ or the target feature representation $M_{\theta_T}(X_T)$ as input and decides whether the input is from the source domain or the target domain, as shown in Fig. 3.1(b). In other words, D_ω is a binary domain classifier. If the accuracy of D_ω is low, we consider $M_{\theta_S}(X_S)$ and $M_{\theta_T}(X_T)$ indistinguishable.

The ADDA method alternately trains the target encoder and the discriminator in two steps. First, the source feature representations with a *source domain label* (say, –1; note that domain labels are not emotion labels) and target feature representations with a *target domain label* (say, +1) are taken as the input to the discriminator, and weights of the discriminator are updated to *minimize* a discriminator loss L_d that aims to promote the accuracy of domain classification. In our approach, the Wasserstein metric [23] is chosen as the loss function to avoid adversarial training from gradient vanishing. Accordingly, the discriminator loss L_d is described by

$$L_d = \frac{1}{N}\sum_{n=1}^{N} D_\omega\left(M_{\theta T}\left({x_T}^{(n)}\right)\right) - D_\omega\left(M_{\theta S}\left({x_S}^{(n)}\right)\right), \qquad (2)$$

where, as defined in (1), N denotes the batch size of source data and target data, $\left\{{x_S}^{(n)}\right\}_{n=1}^{N}$ denotes a batch of X_S, and, similarly, $\left\{{x_T}^{(n)}\right\}_{n=1}^{N}$ denotes a batch of X_S.

Second, the target feature representations with a flipped domain label (e.g. source domain label becomes +1 and target domain label becomes –1) are taken as input to the discriminator, and weights of the target encoder are updated to *maximize* the discriminator loss. Note that weights of the discriminator are fixed in this step to keep the classification ability of the discriminator. In this way, the target encoder can be trained to fool the discriminator. As the learning objective of the target encoder is at odds with the learning objective of the discriminator, we consider the loss function of the target encoder as *adversarial loss* and denote it by *La*,

$$L_a = -\frac{1}{N}\sum_{n=1}^{N} D_\omega\left(M_{\theta_T}\left(x_T{}^{(n)}\right)\right) + D_\omega\left(M_{\theta_S}\left(x_S{}^{(n)}\right)\right). \tag{3}$$

Note that the gradient of the second term with respect to θ_T on the right hand side of (3) becomes zero. Therefore, only the target representations in the first term are input to the discriminator in this step.

We repeat the above two steps until the target encoder model is converged. Besides, as Wasserstein loss is applied under a *K*-Lipschitz constraint, weights of the discriminator are clipped into a compact space with absolute supremum *C* (so *ω* ranges from –*C* to *C*) [23].

doi:10.6342/NTU201802248

Table 3.2. The pseudo code of the adversarial discriminative domain adaptation algorithm

```
Require：$\alpha$, the learning rate. $C$, the clipping parameter. $N$, the batch size. $I$, the number of iterations of the discriminator per generator iteration.
Require：$\omega_0$, weights of an initial discriminator. $\theta_{T_0}$, weights of an initial target encoder. $\theta_S$, weights of the source encoder.
while $\theta_T$ has not converged do
    for $i = 0, \ldots, I$ do
        Sample $\{x_S{}^{(n)}\}_{n=1}^{N}$ a batch of the $X_S$.
        Sample $\{x_T{}^{(n)}\}_{n=1}^{N}$ a batch of the $X_T$.
        $g_\omega \leftarrow \nabla_\omega [\frac{1}{N}\sum_{n=1}^{N} D_\omega\left(M_{\theta T}\left(x_T{}^{(n)}\right)\right) - D_\omega\left(M_{\theta S}\left(x_S{}^{(n)}\right)\right)]$
        $\omega \leftarrow \omega - \alpha \cdot \mathrm{RMSProp}(\omega;\ g_\omega)$
        $\omega \leftarrow \mathrm{clip}(\omega, -C, C)$
    end for
    Sample $\{x_T{}^{(i)}\}_{i=1}^{m}$ a batch of the $X_T$.
    $g_{\theta_T} \leftarrow \nabla_{\theta_T} [-\frac{1}{N}\sum_{n=1}^{N} D_\omega\left(M_{\theta T}\left(x_T{}^{(n)}\right)\right) + D_\omega\left(M_{\theta S}\left(x_S{}^{(n)}\right)\right)]$
    $\theta_T \leftarrow \theta_T - \alpha \cdot \mathrm{RMSProp}(\theta_T;\ g_{\theta_T})$
end while
```

A pseudo code of the adversarial discriminative domain adaptation algorithm is described in Table 3.2. To take advantage of the pre-trained source encoder, we use the source encoder as the initial target encoder.

doi:10.6342/NTU201802248

Chapter 4 NETWORK ARCHITECTURE

The network architecture for pre-training and adaptation is shown in Fig. 4.1. In this section, we first describe the details of the source encoder, the source regressor, and the discriminator. Then, we describe a simple feature fusion method to improve emotion prediction.

Each song is clipped into 29 seconds to fit the smallest song size and resampled to 22,050 Hz. The input data are three types of acoustic features extracted from these song clips. We use a different 2D convolutional neural network (2D-ConvNet) to encode each type of acoustic feature. The first dimension of the filter in the first convolutional layer is

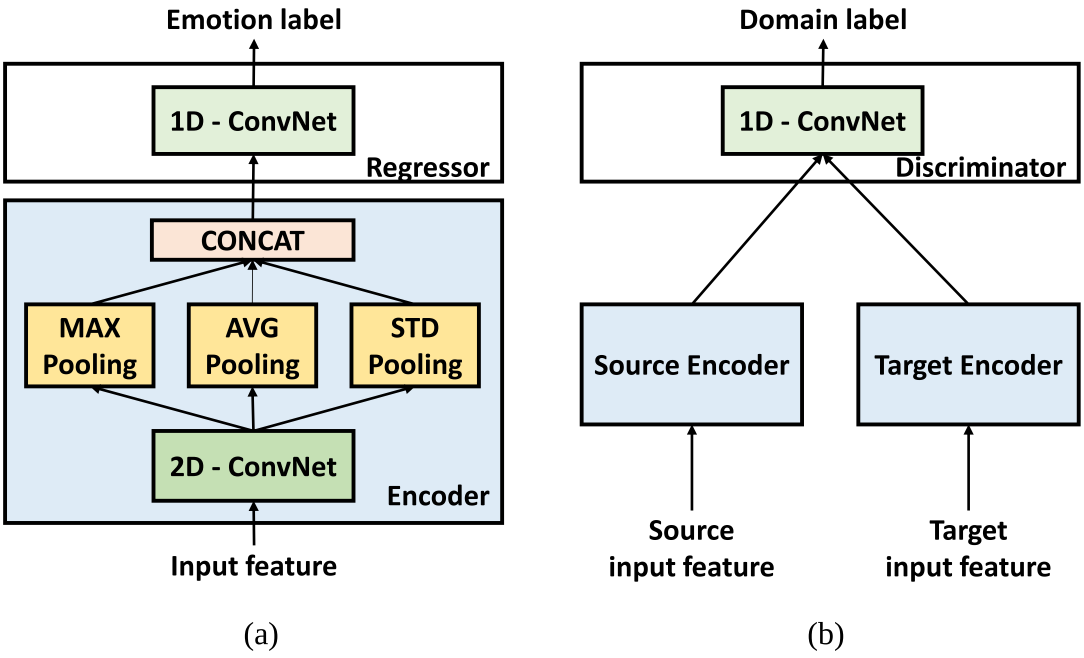


Figure 4.1. Two different network architectures used in our experiments. (a) The network architecture of pre-training MER model. (b) The network architecture of adversarial discriminative domain adaptation.

doi:10.6342/NTU201802248

equal to the number of frequency bins of the input feature, because different frequency bins may carry different information of music emotion. Also, we apply three types of pooling (max pooling, average pooling, and standard deviation pooling) to aggregate the output feature maps of these 2D-ConvNets. Because 2D-ConvNets for the three feature inputs use the same number (128) of filters for each layer, the dimensions (128×3) of the concatenated pooling outputs are the same. The concatenation of the pooling outputs is the input to the source regressor and the discriminator.

4.1 Log-mel-spectrogram Encoder

We compute the log-mel-spectrogram to extract timbre-related features of the songs, as is the case in many previous works. The spectrogram is first computed with a Hanning window of 1024 samples and a 512-sample stride size and then transformed into a 96-bin log-mel-spectrogram. As a result, the dimensions of the log-mel-spectrogram are 96×1249.

The 2D-ConvNet of the log-mel-spectrogram encoder consists of five convolutional layers. The dimensions of the filters are 96×4, 1×4, 1×3, 1×3, and 1×2, and the filter stride sizes are 1×3, 1×2, 1×3, 1×3, and 1×2 for the five layers. Each convolutional layer is followed by a batch normalization and an ELU activation function.

4.2 Pitch Encoder

We apply the pre-trained deep convolutional network proposed by Bitter et al. [24] to extract the pitch salience representation. The goal is to learn the perceived spectral amplitude over time of polyphonic music. Specifically, the harmonic contents are emphasized and the un-pitched or noise contents are de-emphasized to generate the pitch salience representation. Since harmonic summation is usually used to extract pitch content, the network takes harmony-related features extracted by the harmonic constant-Q transform (HCQT) as input. The HCQT generates a time-frequency feature map for each harmonic. The network output has the same size as any harmonic feature map (time-frequency representation).

The frequency dimension of HCQT is partitioned into 360 bins (60 bins per octave for 6 octaves), and the HCQT is computed for 6 harmonic bins using a 512-sample stride size. The resulting 6×360×1249 HCQT feature map is input to the network to generate a 360×1249 pitch salience representation.

Because log-mel-spectrogram and pitch salience representation have the same length in time, we simply change the first dimension of the first filter from 96 to 360 for the 2D-ConvNet and use the same setting for the other filters.

doi:10.6342/NTU201802248

4.3 Autocorrelation-Based Tempogram Encoder

We use the autocorrelation-based tempogram through the tempogram toolbox [25] to extract rhythm-related features. Inspired by chromagram, the toolbox applies the concept of tempogram, which is a time-tempo representation for a given time-dependent signal. We adopt an autocorrelation based method with a 0.2-second stride size to extract a 571-bin tempogram, and the resulting dimensions are 571×142, where the first dimension represents the tempo and the second one represents time.

Because the resulting tempogram feature is relatively small, the 2D-ConvNet of the autocorrelation-based tempogram encoder consists of only three convolutional layers. The dimensions of the filters are 571×4, 1×3, and 1×3, and the filter stride sizes are 1×3, 1×2, and 1×2.

4.4 Regressor and Discriminator

As described earlier, the pooling outputs of the source encoder are concatenated into a 128×3 source representation so that the subsequent network can assign individual weights to the three pooling outputs. The resulting representation is input to a regressor consisting of a three-layer 1D convolutional neural network (1D-ConvNet) to recognize the emotion values. The 1D ConvNet of the regressor has 64, 128, and 256 filters for the three layers, the corresponding dimension of filters are 8, 4, and 2, and the corresponding

doi:10.6342/NTU201802248

filter stride sizes are 4, 2, and 1. The 1D ConvNet is activated by an ELU at each layer. The output feature maps are flattened to 1D and activated by a tanh neuron to predict emotion values ranging from –1 to 1. The same 1D-ConvNet is used for the discriminator except that the last tanh activation neuron is replaced by a linear activation neuron for computing the Wasserstein loss.

4.5 Fusion

Because each predicted emotion label is a single value, our fusion method simply takes the average of the predicted emotion values for MER models that use different input features.

doi:10.6342/NTU201802248

Chapter 5 EXPERIMENTS SETTING

For evaluating the pre-trained MER models, performances were averaged across 10-fold cross validation. As only one dataset was used for training and testing, we called the experiment *within-dataset experiment*. To test if our adaptation method can reduce the domain shift effect, we compared performances between the pre-trained models and the adapted models by averaging performances across 10 segmentations of the target dataset. As datasets used for training and testing were different, we called the experiment *cross-dataset experiment*.

5.1 Datasets

We chose AMG1608 as our source English dataset and CH818 as our target Chinese dataset. The AMG1608 dataset was created by Chen et al. [26]. It consists of 1,608 Western song clips of 30 seconds available on 7digital, a popular music stream service. The valence and arousal emotion were annotated by Americans using Amazon Mechanic Turk (MTurk), which is a crowdsourcing platform, on a two-dimensional space with coordinates ranging from –1 to 1. To ensure the annotation quality, duplicated clips were applied to guarantee the reliability of the annotator. Each clip was annotated by 15–32 annotators.

doi:10.6342/NTU201802248

The CH818 dataset contains 818 Chinese pop song clips released in Taiwan, Hong Kong, and Mainland China. Specifically, each song was clipped into several 30-second segments and predicted emotion values through a pre-trained regression model to choose the most emotional part as stimuli [12]. Each clip was annotated by three Chinese music experts with two independent sliding bars ranging from –10 to 10 for valence and arousal. An annotation instruction and a training session were given before the subjective test to ensure that the annotators fully understand the annotation task. Though the number of annotators is smaller than the AMG1608, the annotations are more consistent. We normalized the emotion annotations so that they are in [–1, 1].

5.2 Training Parameters

The regression model was trained by using the Adam optimizer with $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$. The ADDA was trained by using the RMSProp optimizer with α = 0.001, 2000 epochs, clipping value 0.01, and five iterations of discriminator. The batch size is set to 16 for both trainings.

5.3 Baseline

The method proposed by Hu and Yang [7] that explored cross-dataset generalizability was adopted as the baseline. Three types of single feature input (including

doi:10.6342/NTU201802248

features related to timbre, pitch, and rhythm) and multiple feature inputs were used for comparison. For timbre-related feature, our method using log-mel-spectrogram was compared with the baseline method using dissonance. For pitch-related feature, our method using pitch salience representation feature was compared with the baseline method using log-chromagram. For rhythm-related feature, our method using autocorrelation-based tempogram was compared with the baseline method using autocorrelation-based cyclic tempogram. For multiple-feature inputs, our fusion method using combinations of different feature-predictions was compared with the baseline method using the combined feature set.

doi:10.6342/NTU201802248

Chapter 6 RESULTS AND DISCUSSION

For the within-dataset experiment, the model was trained and tested on AMG1608 by 10-fold cross validation. Our method without adaptation was compared with the baseline method [7] for the within-dataset experiment to evaluate performances of MER models using different features. For the cross-dataset experiment, the model was trained on AMG1608 and tested on 10 segmented subsets of CH818. Our method with adaptation was compared with our method without adaptation and the baseline method for the cross-dataset experiment to examine if the adaptation phase of our method can reduce the effect of domain shift. Also, two metrics were used for the regression performance evaluation. The first metric R^2, which is the square of correlations between predicted values and ground truth values, is a correlation measure. The second metric *RMSE*, which is the root of mean squared error between predicted values and ground truth values, is an absolute-distance measure. Note that performances measured by *RMSE* were showed only for our method because Hu and Yang [7] did not use the metric for evaluation.

6.1 Analysis of Training ADDA

In order to better realize the training process of ADDA, we first showed the performance (measured by R^2) of our method with adaptation during training. Fig. 6.1

doi:10.6342/NTU201802248

shows the comparison of the training regression performance (in R^2) for valence prediction using different features by our method. The model performances are sampled once per 100 epochs for the training curve. For our method with adaptation using different features, the log-mel-spectrogram and pitch salience representation performed better than the autocorrelation-based tempogram in average. Also, the log-mel-spectrogram performed the best in average. Besides, our method with adaptation can perform better than our method without adaptation except the case that the input feature is the autocorrelation-based tempogram.

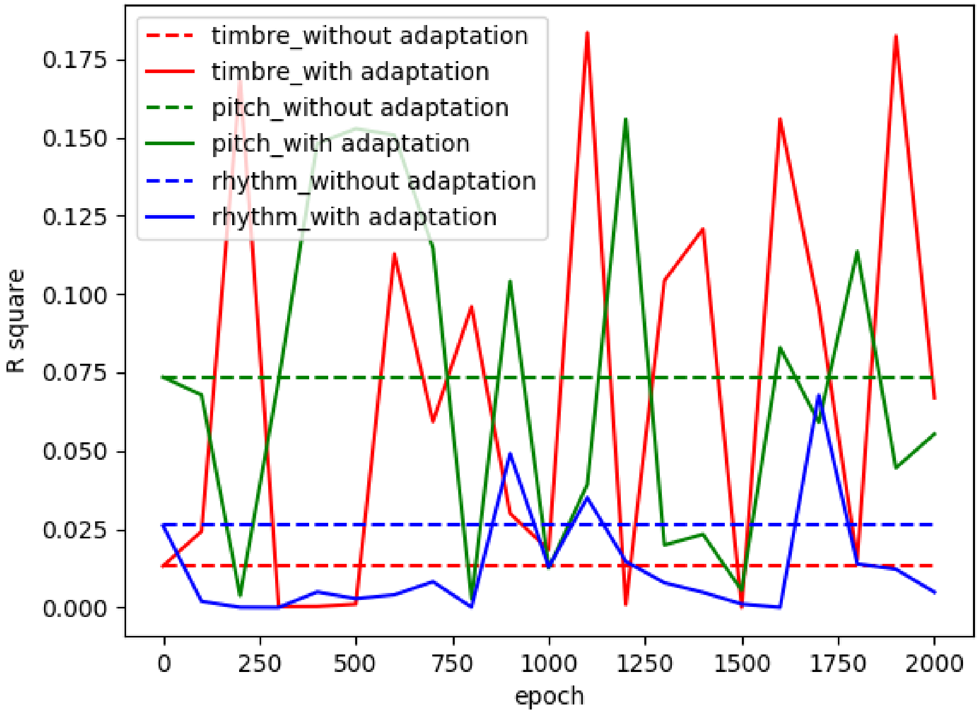


Figure 6.1. Comparison of the training regression performance (measured by R^2) for valence prediction using different features by our method.

doi:10.6342/NTU201802248

Fig. 6.2 shows the comparison of the training regression performance (in R^2) for arousal prediction using different features by our methd. For our method with adaptation using different features, results similar to valence prediction were obtained. Our method with adaptation performed the best for the log-mel-spactrogram and the worst for the autocorrelation-based tempogram. However, we found the arousal prediction did not improve by our adaptation method.

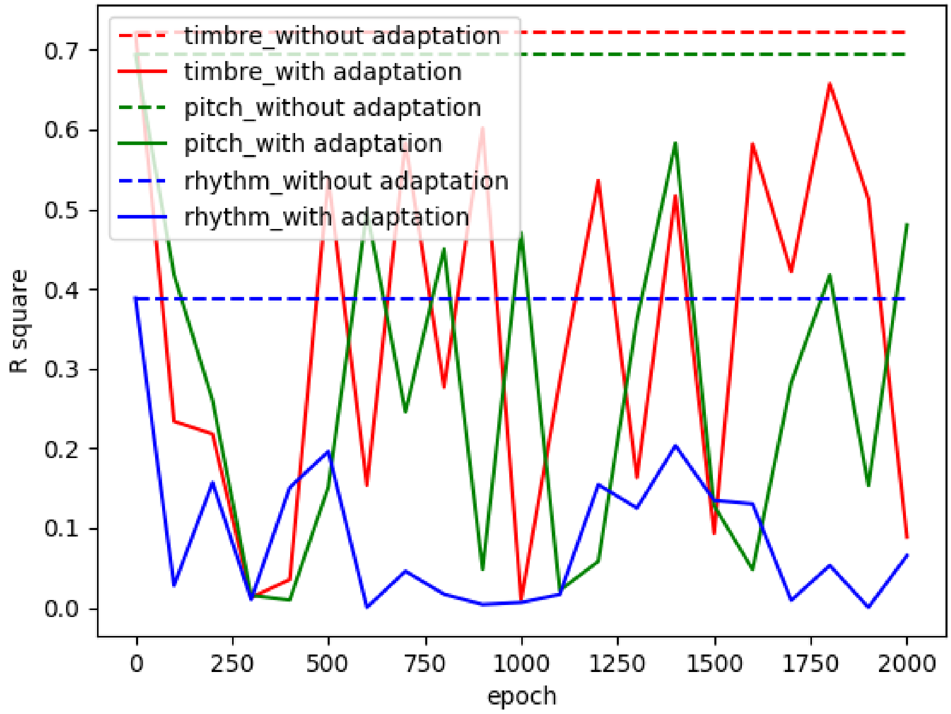


Figure 6.2. Comparison of the training regression performance (measured by R^2) for arousal prediction using different features by our method.

6.2 Within-Dataset Experiment

a) Valence

doi:10.6342/NTU201802248

Table 6.1. Comparison of the within-dataset regression performance (measured by R^2) for valence prediction using single feature.

Methods	Features		
	Timbre	Pitch	Rhythm
Hu and Yang [7]	0.10	0.07	0.07
Our method without adaptation	**0.24**	0.22	0.05

The best performance is expressed in bold.

Table 6.2. The within-dataset regression performance (measured by R^2) for arousal prediction using multiple features.

Method	Features			
	Timbre+ pitch	Timbre+ rhythm	Rhythm+ pitch	Timbre+ pitch+rhythm
Our method without adaptation	0.32	0.22	0.21	0.31

The best performance is expressed in bold.

Table 6.1 shows the comparison of the within-dataset regression performance (measured by R^2) for valence prediction using single feature. Our method without adaptation performed better than the baseline except the case where the input feature is the rhythm-related feature (autocorrelation-based tempogram). Among the three types of input features, our method without adaptation performed the best for the timbre-related feature (log-mel-spectrogram). The result is reasonable as the log-mel-spectrogram has been shown to be an effectiveness-feature in many music related tasks [27].

Table 6.2 shows the within-dataset regression performance (measured by R^2) for valence prediction using multiple features. Our fusion method can perform even better than our method using single feature. Among all the combinations for different feature predictions, fusing predictions for the log-mel-spectrogram and the pitch salience

Table 6.3. Comparison of the within-dataset regression performance (measured by R^2) for arousal prediction using single feature.

Methods	Features		
	Timbre	Pitch	Rhythm
Hu and Yang [4]	0.68	0.57	0.31
Our method without adaptation	0.82	0.68	0.26

The best performance is expressed in bold.

Table 6.4. The within-dataset regression performance (measured by R^2) for arousal prediction using multiple features.

Method	Features			
	Timbre+ pitch	Timbre+ rhythm	Rhythm+ pitch	Timbre+ pitch+rhythm
Our method without adaptation	0.82	0.73	0.65	0.79

The best performance is expressed in bold.

representation (the two best-performing features) led to the best performance (R^2 = 0.32) and outperformed the baseline using the combined feature sets (R^2 = 0.14) [7], including features related to loudness, harmony, and timbre.

b) Arousal

Table 6.3 shows the comparison of the within-dataset regression performance (measured by R^2) for valence prediction using single feature. Similar to valence prediction, our method without adaptation performed better than the baseline except the case where the input feature is the autocorrelation-based tempogram. Among all the features, the autocorrelation-based tempogram performed the worst. Although we know rhythm is much related to arousal in music psychology (e.g. faster songs with higher arousal values), previous studies also found similar results that using rhythm-related feature as input

doi:10.6342/NTU201802248

feature rarely performed well for arousal prediction [6], [7], [28].

Table 6.4 shows the within-dataset regression performance (measured by R^2) for arousal prediction using multiple features. The combination of the log-mel-spectrogram and the pitch salience representation performed the best for our fusion method ($R^2 = 0.82$) among all the other combinations, just the same as the performance of our method using the log-mel-spectrogram alone, and outperformed the baseline using the combined feature sets ($R^2 = 0.73$) [7], including features related to timbre and rhythm. Because using the autocorrelation-based tempogram alone did not performed well, any combination with the autocorrelation-based tempogram for our fusion method degraded the prediction accuracy.

For the within-dataset experiments, our method without adaptation can achieve better performance than the baseline for both valence and arousal prediction. The reason may be the convolutional neural networks used for our method have ability to learn the appropriate feature representation for MER.

6.3 Cross-Dataset Experiment

a) Valence

doi:10.6342/NTU201802248

Table 6.5. Comparison of the cross-dataset regression performance (measured by R^2) for valence prediction using single feature.

Methods	Features		
	Timbre	Pitch	Rhythm
Hu and Yang [4]	0.11	0.07	0.18
Our method without adaptation	0.03	0.08	0.04
Our method with adaptation	0.21	0.18	0.06

The best performance is expressed in bold.

Table 6.6. Comparison of the cross-dataset regression performance (measured by R^2) for valence prediction using multiple features by our method.

Methods	Features			
	Timbre+ pitch	Timbre+ rhythm	Rhythm+ pitch	Timbre+ pitch+rhythm
Without adaptation	0.08	0.05	0.08	0.09
With adaptation	0.22	0.22	0.17	0.23

The best performance is expressed in bold.

Table 6.5 shows the comparison of the cross-dataset regression performance (measured by R^2) for valence prediction using single feature. Though our method without adaptation performed well for the within-dataset valence prediction, performances degraded for cross-dataset valence prediction due to the domain shift effect. Our method with adaptation did improve cross-dataset valence prediction for all the features and performed better than the baselines for the log-mel-spectrogram (R^2 = 0.21) and the pitch salience representation (R^2 = 0.18).

Fable 6.6 shows the comparison of the cross-dataset regression performance (measured by R^2) for valence prediction using multiple features. All the combinations of our fusion method with adaptation (R^2 >= 0.22) were much better than our method with

Table 6.7. Comparison of the cross-dataset regression performance (measured by *RMSE*) for valence prediction using different features by our method.

Methods	Features						
	Timbre	Pitch	Rhythm	Timbre+ pitch	Timbre+ rhythm	Rhythm+ pitch	Timbre+ pitch+rhythm
without adaptation	0.39	0.38	0.40	0.38	0.39	0.38	0.38
with adaptation	0.12	0.38	0.38	0.35	0.35	0.36	0.35

The best performance is expressed in bold.

adaptation using the best-performing single feature (R^2 = 0.21) and were better than the baseline using the combined feature sets (R^2 = 0.21) [7] except the combination of pitch salience representation and autocorrelation-based tempogram. Among these combinations, combined with all the three features (R^2 = 0.23) performed the best. Note that the autocorrelation-based tempogram was helpful for our fusion method, although our method using the feature alone did not perform well.

Table 6.7 shows the comparison of the regression performances (measured by *RMSE*) for valence prediction using different features. Our method with adpatation performed better than our method without adaptation for all the features in general. Among all the features used for our method with adaptation, the log-mel-spectrogram led to the best performance (*RMSE* = 0.12).

For the cross-dataset valence prediction, our method with adaptation performed better than our method without adaptation measured by R^2 in general. Among all the

Table 6.8. Comparison of the cross-dataset regression performance (measured by R^2) for arousal prediction using single feature.

Methods	Features		
	Timbre	Pitch	Rhythm
Hu and Yang [4]	0.66	0.71	0.55
Our method without adaptation	0.72	0.69	0.39
Our method with adaptation	0.73	0.65	0.28

The best performance is expressed in bold.

Table 6.9. Comparison of the cross-dataset regression performance (measured by R^2) for arousal prediction using multiple features by our method.

Methods	Features			
	Timbre+ pitch	Timbre+ rhythm	Rhythm+ pitch	Timbre+ pitch+rhythm
without adaptation	0.74	0.68	0.67	0.74
with adaptation	0.76	0.65	0.49	0.71

The best performance is expressed in bold.

features used for our method with adaption, fusing predictions for the three features performed the best measured by R^2 and performed comparable with other combinations measured by *RMSE*. As a result, we chose all the three features as our model input for valence prediction.

b) Arousal

Table 6.8 shows the comparison of the cross-dataset regression performance (measured by R^2) for arousal prediction using single feature. Our method with adaptation using the log-mel-spectrogram (R^2 = 0.73) performed better than our method without adaptation using the same feature (R^2 = 0.72) and performed better than the baseline using the same feature (R^2 = 0.66). However, our method with adaptation did not perform better

Table 6.10. Comparison of the cross-dataset regression performance (measured by *RMSE*) for arousal prediction using different features by our method.

Methods	Features						
	Timbre	Pitch	Rhythm	Timbre+ pitch	Timbre+ rhythm	Rhythm+ pitch	Timbre+ pitch+rhythm
without adaptation	0.44	0.43	0.39	0.43	0.39	0.39	0.40
with adaptation	0.18	0.42	0.36	0.40	0.31	0.34	0.33

The best performance is expressed in bold.

than our method without adaptation for the pitch salience representation and the autocorrelation-based tempogram. The reason may be that arousal is relatively more generalizable across datasets as previous studies shown [12], [13]. Therefore, unsupervised adaptation could not improve the result.

Table 6.9 shows the comparison of the cross-dataset regression performance (measured by R^2) for arousal prediction using multiple features. Our fusion method with adaptation fusing predictions for the log-mel-spectrogram and the pitch salience representation performed the best (R^2 = 0.76) and performed better than the baseline using the combined feature sets (R^2 = 0.68) [7].

Table 6.10 shows the comparison of the regression performances (measured by *RMSE*) for arousal prediction using different features. Similar to valence prediction, our method with adpatation performed better than our method without adaptation for all the features in general. Among all the features used for our method with adaptation, the log-mel-spectrogram led to the best performance (*RMSE* = 0.18).

For the cross-dataset arousal prediction, the combination for the log-mel-spectrogram and the pitch salience representation performed the best measured by R^2 and performed the second-best measured by *RMSE*. As a result, we chose the log-mel-

doi:10.6342/NTU201802248

spectrogram and the pitch salience representation as our model input for arousal prediction.

doi:10.6342/NTU201802248

Chapter 7 CONCLUSION

This study has explored cross-dataset adaptation of music emotion recognition by adversarial discriminative domain adaptation (ADDA). For cross-dataset experiment, the results show that our method perform better than Hu and Yang [7] for both valence prediction and arousal prediction. Also, our adaptation method do improve our pre-training method for valence prediction but not for arousal prediction, possibly because arousal prediction is more easily generalizable across datasets. For future work, we want to experiment on a small number of labeled target data for few-shot learning [29], [30], to analyze what are the musical features that are actually adapted by ADDA, and to experiment with other domain adaptation methods. Moreover, the present method only accounts for the cultural differences in music features, but not for the cultural differences in emotion perception. This is a subject of future work as well.

doi:10.6342/NTU201802248

REFERENCES

[1] Y. H. Yang and H. H. Chen, "Machine recognition of music emotion: A review," *ACM Trans. Intelligent Sys. Technology*, vol. 3, no. 3, p. 40, May 2012.

[2] M. Barthet, G. Fazekas, and M. Sandler, "Music emotion recognition: From content to context-based models," in *Int. Symposium on Comput. Music Modeling and Retrieval*, Berlin, Germany: Springer-Verlag, 2013, pp. 228–252.

[3] A. Aljanaki, Y. H. Yang, and M. Soleymani, "Emotion in music task at mediaeval 2014," in *Proc. MediaEval Workshop*, Barcelona, Spain, vol. 1263, 2014.

[4] X. Hu, and Y. H. Yang, "The mood of Chinese Pop music: Representation and recognition," *Journal of the Association for Information Science and Technology*, vol. 68, no. 8, pp. 1899–1910, 2017.

[5] X. Hu and J. H. Lee, "A cross-cultural study of music mood perception between American and Chinese listeners," in *Int. Soc. Music Information Retrieval*, 2012, pp. 535–540.

[6] K. Kosta, Y. Song, G. Fazekas, and M. B. Sandler, "A study of cultural dependence of perceived mood in Greek music," in *Int. Soc. Music Information Retrieval*, 2013, pp. 1–6.

[7] X. Hu, and Y. H. Yang, “Cross-dataset and cross-cultural music mood prediction: A case on Western and Chinese Pop songs,” in *IEEE Trans. Affective Comput.*, vol. 8, no. 2, pp. 228–240, 2017.

[8] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, “Adversarial discriminative domain adaptation,” in *Comput. Vision and Pattern Recognition*, vol. 1, no. 2, p. 4, July 2017.

[9] G. Csurka, “Domain adaptation for visual applications: A comprehensive survey,” *arXiv preprint arXiv:1702.05374*, 2017.

[10] E. Coutinho, J. Deng, and B. Schuller, “Transfer learning emotion manifestation across music and speech,” in *Int. Joint Conf. Neural Networks*, July 2014, pp. 3592–3598.

[11] X. Hu and Y. H. Yang, “A study on cross-cultural and cross-dataset generalizability of music mood regression models,” in Int. Sound Music Comput. Conf., 2014, pp. 1149–1155.

[12] Y. H. Yang and X. Hu, “Cross-Cultural music mood classification: A comparison on English and Chinese songs,” in *Int. Soc. Music Information Retrieval*, Oct. 2012, pp. 19–24.

doi:10.6342/NTU201802248

[13] T. Eerola, “Are the emotions expressed in music genre-specific? An audio-based evaluation of datasets spanning classical, film, pop and mixed genres,” *Journal of New Music Research*, vol. 40, no. 4, pp. 349–366, 2011.

[14] M. Long, Y. Cao, J. Wang, and M. I. Jordan, “Learning transferable features with deep adaptation networks,” *arXiv preprint arXiv:1502.02791*, 2015.

[15] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, “Deep domain confusion: Maximizing for domain invariance,” *arXiv preprint arXiv:1412.3474*, 2014.

[16] B. Sun, and K. Saenko, “Deep coral: Correlation alignment for deep domain adaptation,” in *European Conf. Comput. Vision*, Springer, Cham, Oct. 2016, pp. 443–450.

[17] B. Sun, J. Feng, and K. Saenko, “Return of Frustratingly Easy Domain Adaptation,” in *Association for the Advancement of Artificial Intelligence*, vol. 6, no. 7, p. 8, Feb. 2016.

[18] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li, “Deep reconstruction-classification networks for unsupervised domain adaptation,” in *European Conf. Comput. Vision*, Springer, Cham., Oct. 2016, pp. 597–613.

doi:10.6342/NTU201802248

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, …, and Y. Bengio, “Generative adversarial nets”, in *Advances in neural information process. sys.*, 2014, pp. 2672–2680.

[20] J. Hoffman, E. Tzeng, T. Park, J. Y. Zhu, P. Isola, K. Saenko, ... and T. Darrell, “Cycada: Cycle-consistent adversarial domain adaptation,” *arXiv preprint arXiv:1711.03213*, 2017.

[21] P. Haeusser, T. Frerix, A. Mordvintsev, and D. Cremers, “Associative domain adaptation,” in *Int. Conf. Comput. Vision*, vol. 2, no. 5, p. 6, Oct. 2017.

[22] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa, “Generate to adapt: Aligning domains using generative adversarial networks,” *arXiv e-prints, abs/1704.01705*, 2017.

[23] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein gan,” *arXiv preprint arXiv:1701.07875*, 2017.

[24] R. M. Bittner, B. McFee, J. Salamon, P. Li, and J. P. Bello, “Deep salience representations for f0 estimation in polyphonic music,” in *Int. Soc. Music Information Retrieval*, Suzhou, China, Oct. 2017, pp. 23–27.

[25] P. Grosche, M. Muller, and F. Kurth, “Cyclic tempogram—a midlevel tempo representation for music signals,” in *IEEE Trans. Acoustics Speech Signal Process.*, 2010, pp. 5522–5525.

doi:10.6342/NTU201802248

[26] Y. A. Chen, Y. H. Yang, J. C. Wang, and H. Chen, “The AMG1608 dataset for music emotion recognition,” in *IEEE Trans. Acoustics, Speech and Signal Process.*, Apr. 2015, pp. 693–697.

[27] Y. S. Huang, S. Y. Chou, and Y. H. Yang, “Pop Music Highlighter: Marking the Emotion Keypoints,” *arXiv preprint arXiv:1802.10495*, 2018.

[28] Y. H. Yang, and H. H. Chen, “Predicting the distribution of perceived emotions of a music signal for content retrieval,” in *IEEE Trans. Audio, Speech, and Language Process.*, vol. 19, no. 7, pp. 2184–2196, Sep. 2011.

[29] S. Ravi, and H. Larochelle, “Optimization as a model for few-shot learning”, in *Int. Conf. Learning Representations*, 2017.

[30] J. Snell, K. Swersky, and R. Zemel, “Prototypical networks for few-shot learning,” in *Advances in Neural Information Process. Sys.*, pp. 4077-4087, 2017.