TRANSFER LEARNING FOR MUSIC CLASSIFICATION AND REGRESSION TASKS

Keunwoo Choi, György Fazekas, Mark Sandler
Centre for Digital Music
Queen Mary University of London, London, UK
keunwoo.choi@qmul.ac.uk

Kyunghyun Cho
Center for Data Science
New York University, New York, NY, USA
kyunghyun.cho@nyu.edu

ABSTRACT

In this paper, we present a transfer learning approach for music classification and regression tasks. We propose to use a *pre-trained convnet feature*, a concatenated feature vector using the activations of feature maps of multiple layers in a trained convolutional network. We show how this convnet feature can serve as general-purpose music representation. In the experiments, a convnet is trained for music tagging and then transferred to other music-related classification and regression tasks. The convnet feature outperforms the baseline MFCC feature in all the considered tasks and several previous approaches that are aggregating MFCCs as well as low- and high-level music features.

1. INTRODUCTION

In the field of machine learning, transfer learning is often defined as *re-using parameters* that are trained on a *source* task for a *target* task, aiming to transfer knowledge between the domains. A common motivation for transfer learning is the lack of sufficient training data in the target task. When using a neural network, by transferring pre-trained weights, the number of trainable parameters in the target-task model can be significantly reduced, enabling effective learning with a smaller dataset.

A popular example of transfer learning is semantic image segmentation in computer vision, where the network utilises rich information, such as basic shapes or prototypical templates of objects, that were captured when trained for image classification [37]. Another example is pre-trained word embeddings in natural language processing. Word embedding, a vector representation of a word, can be trained on large datasets such as Wikipedia [35] and adopted to other tasks such as sentiment analysis [27].

 This work has been part funded by FAST IMPACt EPSRC Grant EP/L019981/1 and the European Commission H2020 research and innovation grant AudioCommons (688382). Mark Sandler acknowledges the support of the Royal Society as a recipient of a Wolfson Research Merit Award. Kyunghyun Cho thanks the support by eBay, TenCent, Facebook, Google and NVIDIA.

There have been several works on transfer learning in Music Information Retrieval (MIR). Hamel et al. proposed to directly learn music features using linear embedding [57] of mel-spectrogram representations and genre/similarity/tag labels [20]. Oord et al. outlines a large-scale transfer learning approach, where a multi-layer perceptron is combined with the spherical K-means algorithm [16] trained on tags and play-count data [54]. After training, the weights are transferred to perform genre classification and auto-tagging with smaller datasets. In music recommendation, Choi et al. used the weights of a convolutional neural network for feature extraction in playlist generation [10], while Liang et al. used a multi-layer perceptron for feature extraction of content-aware collaborative filtering [29].

2. TRANSFER LEARNING FOR MUSIC

In this section, our proposed transfer learning approach is described. A convolutional neural network (convnet) is designed and trained for a source task, and then, the network with trained weights is used as a feature extractor for target tasks. The schematic of the proposed approach is illustrated in Figure 1.

2.1 Convolutional Neural Networks for Music Tagging

We choose music tagging as a source task because *i)* large training data is available and *ii)* its rich label set covers various aspects of music, e.g., *genre*, *mood*, *era*, and *instrumentations*. In the source task, a mel-spectrogram ($\mathbf{X}$), a two-dimensional representation of music signal, is used as the input to the convnet. The mel-spectrogram is selected since it is psychologically relevant and computationally efficient. It provides a mel-scaled frequency representation which is an effective approximation of human auditory perception [36] and typically involves compressing the frequency axis of short-time Fourier transform representation (e.g., 257/513/1025 frequency bins to 64/96/128 Mel-frequency bins). In our study, the number of mel-bins is set to 96 and the magnitude of mel-spectrogram is mapped to decibel scale ($\log_{10}\mathbf{X}$), following [8] since it is also shown to be crucial in [7].

In the proposed system, there are five layers of convolutional and sub-sampling in the convnet as shown in Figure 1. This convnet structure with 2-dimensional 3×3 kernels and 2-dimensional convolution, which is often called *Vgnet* [44], is expected to learn hierarchical time-frequency

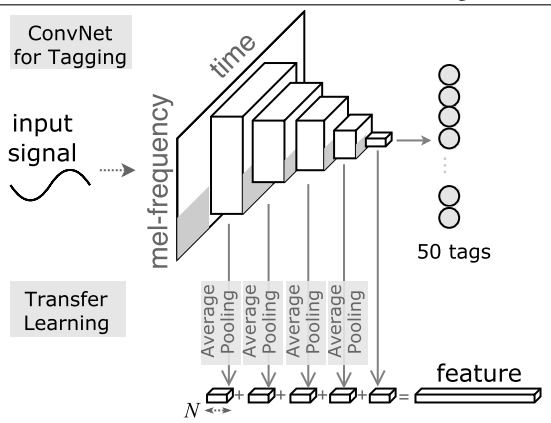

Figure 1: A block diagram of the training and feature extraction procedures. Exponential linear unit (ELU) is used as an activation function in all convolutional layers [15]. Max-pooling of (2, 4), (4, 4), (4, 5), (2, 4), (4, 4) is applied after every convolutional layer respectively. In all the convolutional layers, the kernel sizes are (3, 3), numbers of channels N is 32, and Batch normalisation is used [24]. The input has a single channel, 96-mel bins, and 1360 temporal frames. After training, the feature maps from 1st–4th layers are subsampled using average pooling while the feature map of 5th layer is used as it is, since it is already scalar (size 1×1). Those 32-dimensional features are concatenated to form a *convnet feature*.

patterns. This structure was originally proposed for visual image classification and has been found to be effective and efficient in music classification [1] [11].

2.2 Representation Transfer

In this section, we explain how features are extracted from a pre-trained convolutional network. In the remainder of the paper, this feature is referred to as *pre-trained convnet feature*, or simply *convnet feature*.

It is already well understood how deep convnets learn *hierarchical features* in visual image classification [58]. By convolution operations in the forward path, lower-level features are used to construct higher-level features. Subsampling layers reduce the size of the feature maps while adding local invariance. In a deeper layer, as a result, the features become more invariant to (scaling/location) distortions and more relevant to the target task.

This type of hierarchy also exists when a convnet is trained for a music-related task. Visualisation and sonification of convnet features for music genre classification has shown the different levels of hierarchy in convolutional layers [13], [9].

Such a hierarchy serves as a motivation for the proposed transfer learning. Relying solely on the last hidden layer may not maximally extract the knowledge from a pre-trained network. For example, low-level information such as tempo, pitch, (local) harmony or envelop can be captured in early layers, but may not be preserved in deeper layers due to the constraints that are introduced by the network structure: aggregating local information by discarding less-relevant information in subsampling. For the same reason, deep scattering networks [6] and a convnet for music tagging introduced in [28] use multi-layer representations.

Based on this insight, we propose to use not only the activations of the final hidden layer but also the activations of (up to) *all* intermediate layers to find the most effective representation for each task. The final feature is generated by concatenating these features as demonstrated in Figure 1, where all the five layers are concatenated to serve as an example.

Given five layers, there are $\sum_{n=1}^{5} {}_5C_n = 31$ strategies of layer-wise combination. In our experiment, we perform a nearly exhaustive search and report all results. We designate each strategy by the indices of layers employed. For example, a strategy named '`135`' refers to using a $32 \times 3 = 96$-dimensional feature vector that concatenates the first, third, and fifth layer convnet features.

During the transfer, average-pooling is used for the 1st–4th layers to reduce the size of feature maps to 1×1 as illustrated in Figure 1. Averaging is chosen instead of `max` pooling because it is more suitable for summarising the global statistics of large regions, as done in the last layer in [30]. Max-pooling is often more suitable for capturing the existence of certain patterns, usually in small and local regions [2].

Lastly, there have been works suggesting random-weights (deep) neural networks including deep convnet can work well as a feature extractor [22] [59] (Not identical, but a similar approach is transferring knowledge from an irrelevant domain, e.g., visual image recognition, to music task [19].) We report these results from random convnet features and denote it as *random convnet feature*. Assessing performances of random convnet feature will help to clarify the contributions of the pre-trained knowledge transfer versus the contributions of the convnet structure and nonlinear high-dimensional transformation.

2.3 Classifiers and Regressors of Target Tasks

Variants of support vector machines (SVMs) [45, 50] are used as a classifier and regressor. SVMs work efficiently in target tasks with small training sets, and outperformed K-nearest neighbours in our work for all the tasks in a preliminary experiment. Since there are many works that use hand-written features and SVMs, using SVMs enables us to focus on comparing the performances of features.

3. PREPARATION

3.1 Source Task: Music Tagging

In the source task, 244,224 preview clips of the Million Song Dataset [5] are used (201,680/12,605/25,940 for training/validation/test sets respectively) with top-50 *last.fm* tags including genres, eras, instrumentations, and moods. Mel-spectrograms are extracted from music signals in real-time on the GPU using *Kapre* [12]. Binary cross-entropy is used as the loss function during training.

[1] For more recent information on kernel shapes for music classification, please see [40].

[2] Since the average is affected by zero-padding which is applied to signals that are shorter than 29 seconds, those signals are repeated to create 29-second signals. This only happens in Task 5 and 6 in the experiment.

Task	Dataset name	#clips	Metric	#classes
T1. Ballroom dance genre classification	Extended ballroom [32]	4,180	Accuracy	13
T2. Genre classification	Gtzan genre [53]	1,000	Accuracy	10
T3. Speech/music classification	Gtzan speech/music [52]	128	Accuracy	2
T4. Emotion prediction	EmoMusic (45-second) [46]	744	Coefficient of determination (r^2)	N/A (2-dimensional)
T5. Vocal/non-vocal classification	Jamendo [41]	4,086	Accuracy	2
T6. Audio event classification	Urbansound8K [42]	8,732	Accuracy	10

Table 1: The details of the six tasks and datasets used in our transfer learning evaluation.

The ADAM optimisation algorithm [25] is used for accelerating stochastic gradient descent. The convnet achieves 0.849 AUC-ROC score (Area Under Curve - Receiver Operating Characteristic) on the test set. We use the *Keras* [14] and *Theano* [51] frameworks in our implementation.

3.2 Target Tasks

Six datasets are selected to be used in six target tasks. They are summarised in Table 1.

- Task 1: The Extended ballroom dataset consists of specific Ballroom dance sub-genres.
- Task 2: The Gtzan genre dataset has been extremely popular, although some flaws have been found [48].
- Task 3: The dataset size is smaller than the others by an order of magnitude.
- Task 4: Emotion predition on the arousal-valence plane. We evaluate arousal and valence separately. We trim and use the first 29-second from the 45-second signals.
- Task 5. Excerpts are subsegments from tracks with binary labels (*'vocal'* and *'non-vocal'*). Many of them are shorter than 29s. This dataset is provided for benchmarking frame-based vocal detection while we use it as a pre-segmented classification task, which may be easier than the original task.
- Task 6: This is a non-musical task. For example, the classes include *air conditioner*, *car horn*, and *dog bark*. All excerpts are shorter than 4 seconds.

3.3 Baseline Feature and Random Convnet Feature

As a baseline feature, the means and standard deviations of 20 Mel-Frequency Cepstral Coefficients (MFCCs), and their first and second-order derivatives are used. In this paper, this baseline feature is called *MFCCs* or *MFCC vectors*. MFCC is chosen since it has been adopted in many music information retrieval tasks and is known to provide a robust representation. *Librosa* [34] is used for MFCC extraction and audio processing.

The random convnet feature is extracted using the identical convnet structure of the source task and after random weights initialisation with a normal distribution [21] but without a training.

4. EXPERIMENTS

4.1 Configurations

For Tasks 1-4, the experiments are done with 10-fold cross-validation using stratified splits. For Task 5, pre-defined training/validation/test sets are used. The experiment on Task 6 is done with 10-fold cross-validation without replacement to prevent using the sub-segments from the same recordings in training and validation. The SVM parameters are optimised using grid-search based on the validation results. Kernel type/bandwidth of radial basis function and the penalty parameter are selected from the ranges below:

- Kernel type: [*linear*, *radial*]
  - Bandwidth γ in radial basis function : $[1/2^3, 1/2^5, 1/2^7, 1/2^9, 1/2^{11}, 1/2^{13}, 1/N_f]$
- Penalty parameter C : $[0.1, 2.0, 8.0, 32.0]$

A radial basis function is $\exp(-\gamma|x - x'|^2)$, and γ and N_f refer to the radial kernel bandwidth and the dimensionality of feature vector respectively. With larger C, the penalty parameter or regularisation parameter, the loss function gives more penalty to misclassified items and vice versa. We use *Scikit-learn* [38] for these target tasks. The code for the data preparation, experiment, and visualisation are available on GitHub [3] .

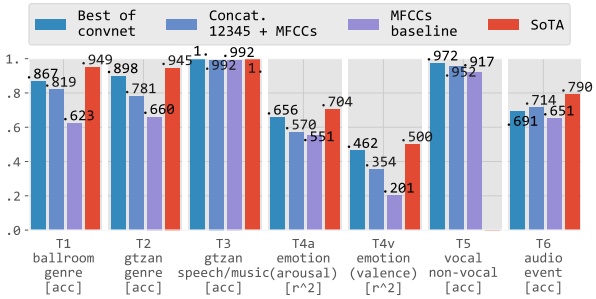

Figure 2: Summary of performances of the convnet feature (blue), MFCCs (purple), and state-of-the-art (red) for Task 1-6 (State-of-the-art of Task 5 does not exist).

4.2 Results and Discussion

Figure 2 shows a summary of the results. The scores of the *i)* best performing convnet feature, *ii)* concatenating '12345' [4] convnet feature and MFCCs, *iii)* MFCC feature, and *iv)* state-of-the-art algorithms for all the tasks.

In all the six tasks, the majority of convnet features outperforms the baseline feature. Concatenating MFCCs

[3] https://github.com/keunwoochoi/transfer_learning_music

[4] Again, '12345' refers to the convnet feature that is concatenated from 1st–5th layers. For another example, '135' means concatenating the features from first, third, and fifth layers.

with '12345' convnet feature usually does not show improvement over a pure convnet feature except in Task 6, audio event classification. Although the reported state-of-the art is typically better, almost all methods rely on musical knowledge and hand-crafted features, yet our features perform competitively. An in-depth look at each task is therefore useful to provide insight.

In the following subsections, the details of each task are discussed with more results presented from (almost) exhaustive combinations of convnet features as well as random convnet features at all layers. For example, in Figure 3, the scores of 28 different convnet feature combinations are shown with blue bars. The narrow, grey bars next to the blue bars indicate the scores of random convnet features. The other three bars on the right represent the scores of the concatenation of '12345' + MFCC feature, MFCC feature, and the reported state-of-the-art methods respectively. The rankings within the convnet feature combinations are also shown *in* the bars where top-7 and lower-7 are highlighted.

We only briefly discuss the results of random convnet features here. The best performing random convnet features do not outperform the best-performing convnet features in any task. In most of the combinations, convnet features outperformed the corresponding random convnet features, although there are few exceptions. However, random convnet features also achieved comparable or even better scores than MFCCs, indicating *i)* a significant part of the strength of convnet features comes from the network structure itself, and *ii)* random convnet features can be useful especially if there is not a suitable source task.

4.2.1 Task 1. Ballroom Genre Classification

Figure 3 shows the performances of different features for Ballroom dance classification. The highest score is achieved using the convnet feature '123' with 86.7% of accuracy. The convnet feature shows good performances, even outperforming some previous works that explicitly use rhythmic features.

The result clearly shows that low-level features are crucial in this task. All of the top-7 strategies of convnet feature include the *second* layer, and 6/7 of them include the *first* layer. On the other hand, the lower-7 are ['5', '4', '3', '45', '35', '2', '25'], none of which includes the first layer. Even '1' achieves a reasonable performance (73.8%).

The importance of low-level features is also supported by known properties of this task. The ballroom genre labels are closely related to rhythmic patterns and tempo [32] [49]. However, there is no label directly related to tempo in the source task. Moreover, deep layers in the proposed structure are conjectured to be mostly invariant to tempo. As a result, high-level features from the fourth and fifth layers poorly contribute to the task relative to those from the first, second, and third layers.

The state-of-the-art algorithm which is also the only algorithm that used the same dataset due to its recent release uses *2D scale transform*, an alternative representation of music signals for rhythm-related tasks [33], and

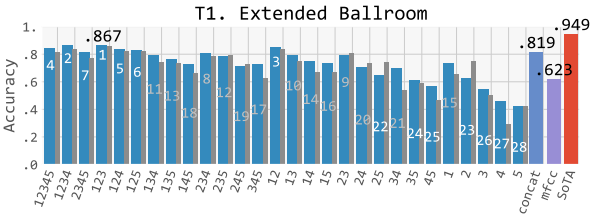

Figure 3: Performances of Task 1 - Ballroom dance genre classification of convnet features (with random convnet features in grey), MFCCs, and the reported state-of-the-art method. (Note the exception that the SoTA is reported in weighted average recall.)

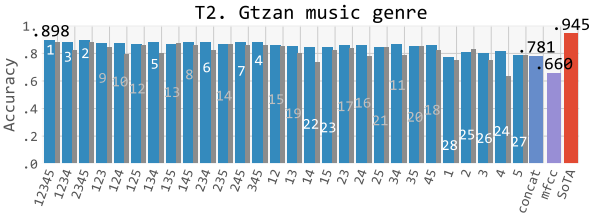

Figure 4: Performances of Task 2 - Gtzan music genre classification of convnet features (with random convnet features in grey), MFCCs, and the reported state-of-the-art method.

reports 94.9% of weighted average recall. For additional comparisons, there are several works that use the Ballroom dataset [18]. This has 8 classes and it is smaller in size than the Extended Ballroom dataset (13 classes). Laykartsis and Lerch [31] combines beat histogram and timbre features to achieve 76.7%. Periodicity analysis with SVM classifier in Gkiokas et al. [17] respectively shows 88.9%/85.6 - 90.7%, before and after feature selection.

4.2.2 Task 2. Gtzan Music Genre Classification

Figure 4 shows the performances on Gtzan music genre classification. The convnet feature shows 89.8% while the concatenated feature and MFCCs respectively show only 78.1% and 66.0% of accuracy. Although there are methods that report accuracies higher than 94.5%, we set 94.5% as the state-of-the-art score following the dataset analysis in [48], which shows that the perfect score cannot surpass 94.5% considering the noise in the Gtzan dataset.

Among a significant number of works that use the Gtzan music genre dataset, we describe four methods in more detail. Three of them use an SVM classifier, which enables us to focus on the comparison with our feature. Arabi and Lu [1] is most similar to the proposed convnet features in a way that it combines low-level and high-level features and shows a similar performance. Beniya et al. [4] and Huang et al. [23] report the performances with many low-level features before and after applying feature selection algorithms. Only the latter outperforms the proposed method and only after feature selection.

- Arabi and Lu [1] uses not only low-level features such as {spectral centroid/flatness/roll-off/flux}, but also high-level musical features such as {beat, chord distribution and chord progressions}. The best combination of the features shows 90.79% of accuracy.
- Beniya et al. [4] uses a particularly rich set of statistics such as {mean, standard deviation, skewness, kurtosis,

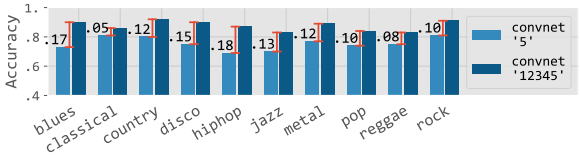

Figure 5: Comparison of per-label results of two convnet feature strategies, '12345' and '5' for Gtzan music genre classification. Numbers denote the differences of scores.

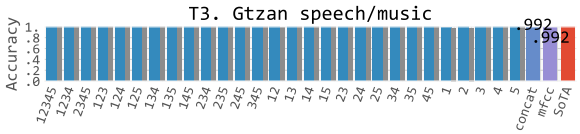

Figure 6: Performances of Task 3 - Speech/music classification of convnet features (with random convnet features in grey), MFCCs, and the reported state-of-the-art method. All scores of convnet features and SoTA are 1.0 and omitted in the plot.

covariance} of many low-level features including {RMS energy, attack, tempo, spectral features, zero-crossing, MFCC, dMFCC, ddMFCC, chromagram peak and centroid}. The feature vector dimensionality is reduced by MRMR (max-relevance and min-redundancy) [39] to obtain the highest classification accuracy of 87.9%.

- Huang et al. [23] adopts another feature selection algorithm, self-adaptive harmony search [55]. The method uses statistics such as {mean, standard deviation} of many features including {energy , pitch, and timbral features} and their derivatives. The original 256-dimensional feature achieved 84.3% of accuracy which increases to 92.2% and 97.2% after feature selection.
- Reusing AlexNet [26], a pre-trained convnet for visual image recognition achieved 78% of accuracy [19].

In summary, the convnet feature achieves better performance than many approaches which use extensive music feature sets without feature selection as well as some of the approaches with feature selection. For this task, it turns out that combining features from all layers is the best strategy. In the results, '12345', '2345', and '1234' are three best configurations, and all of the top-7 scores are from those strategies that use more than three layers. On the contrary, all lower-7 scores are from those with only 1 or 2 layers. This is interesting since the majority (7/10) of the target labels already exists in source task labels, by which it is reasonable to assume that the necessary information can be provided only with the last layer for those labels. Even in such a situation, however, low-level features contribute to improving the genre classification performance [5] .

Among the classes of target task, *classical* and *disco*, *reggae* do not exist in the source task classes. Based on this, we consider two hypotheses, *i)* the performances of those three classes may be lower than the others, *ii)* low-level features may play an important role to classify them since high-level feature from the last layer may be biased to the other 7 classes which exist in the source task. However, both hypotheses are rebutted by comparing the performances for each genres with convnet feature '5' and

[5] On the contrary, in Task 5 - music emotion classification, high-level feature plays a dominant role (see Section 4.2.4).

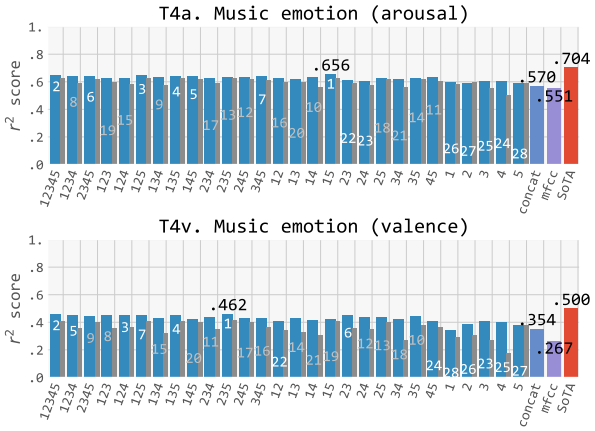

Figure 7: Performances of Task 4a (arousal) and 4v (valence) - Music emotion prediction of convnet features (with random convnet features in grey), MFCCs, and the reported state-of-the-art method.

'12345' as in Figure 5. First, with '5' convnet feature, *classical* shows the highest accuracy while both *disco* and *reggae* show accuracies around the average accuracy reported over the classes. Second, aggregating early-layer features affects all the classes rather than the three omitted classes. This suggests that the convnet features are not strongly biased towards the genres that are included in the source task and can be used generally for target tasks with music different from those genres.

4.2.3 *Task 3. Gtzan Speech/music Classification*

Figure 6 shows the accuracies of convnet features, baseline feature, and state-of-the-art [47] with low-level features including MFCCs and sparse dictionary learning for Gtzan music/speech classification. A majority of the convnet feature combinations achieve 100% accuracy. MFCC features achieve 99.2%, but the error rate is trivial (0.8% is one sample out of 128 excerpts).

Although the source task is only about music tags, the pre-trained feature in any layer easily solved the task, suggesting that the nature of music and speech signals in the dataset is highly distinctive.

4.2.4 *Task 4. Music Emotion Prediction*

Figure 7 shows the results for music emotion prediction (Task 4). The best performing convnet features achieve 0.633 and 0.415 r^2 scores on arousal and valence axes respectively.

On the other hand, the state-of-the-art algorithm reports 0.704 and 0.500 r^2 scores using music features with a recurrent neural network as a classifier [56] that uses 4,777 audio features including many functionals (such as *quantiles, standard deviation, mean, inter peak distances*) of *12 chroma features*, *loudness*, *RMS Energy*, *zero crossing rate*, *14 MFCCs*, *spectral energy*, *spectral roll-off*, etc.

For the prediction of arousal, there is a strong dependency on the last layer feature. All top-7 performances are from the feature vectors that include the fifth layer. The first layer feature also seems important, since all of the top-5 strategies include the first and fifth layer features. For

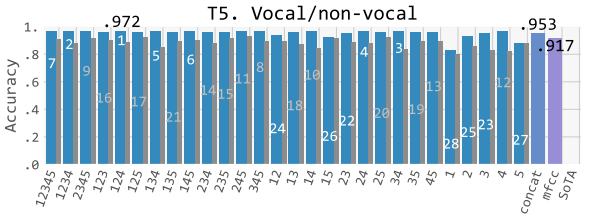

Figure 8: Performances of Task 5 - Vocal detection of convnet features (with random convnet features in grey) and MFCCs.

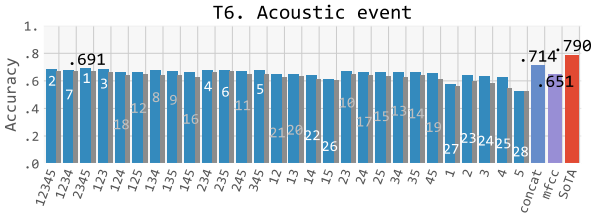

Figure 9: Performances of Task 6 - Acoustic event detection of convnet features (with random convnet features in grey), MFCCs, and the reported state-of-the-art method.

valence prediction, the third layer feature seems to be the most important one. The third layer is included in all of the top-6 strategies. Moreover, '3' strategy was found to be best performing among strategies with single layer feature.

To summarise the results, the predictions of arousal and valence rely on different layers, for which they should be optimised separately. In order to remove the effect of the choice of a classifier and assess solely the effect of features, we compare our approach to the baseline method of [56] which is based on the same 4,777 features with SVM, not a recurrent neural network. The baseline method achieves .541 and .320 r^2 scores respectively on arousal and valence, both of which are lower than those achieved by using the proposed convnet feature. This further confirms the effectiveness of the proposed convnet features.

4.2.5 Task 5. Vocal/non-vocal Classification

Figure 8 presents the performances on vocal/non-vocal classification using the Jamendo dataset [41]. There is no known state-of-the-art result, as the dataset is usually used for *frame-based* vocal *detection/segmentation*. Pre-segmented *Excerpt classification* is the task we formulate in this paper. For this dataset, the fourth layer plays the most important role. All the 14 combinations that include the fourth layer outperformed the other 14 strategies without the fourth layer.

4.2.6 Task 6. Acoustic Event Detection

Figure 9 shows the results on acoustic event classification using Urbansound8K dataset [42]. Since this is not a music-related task, there are no common tags between the source and target tasks, and therefore the final-layer feature is not expected to be useful for the target task.

The strategy of concatenating '12345' convnet features and MFCCs yields the best performance. Among convnet features, '2345', '12345', '123', and '234' achieve good accuracies. In contrast, those with only one or two layers do not perform well. We were not able to observe any particular dependency on a certain layer.

Since the convnet features are trained on music, they do not outperform a dedicated convnet trained for the target task. The state-of-the-art method is based on a deep convolutional neural network with data augmentation [43]. Without augmenting the training data, the accuracy of convnet in the same work is reported to be 74%, which is still higher than our best result (71.4%). [6]

The convnet feature still shows better results than conventional audio features, demonstrating its versatility even for non-musical tasks. The method in [42] with {*minimum, maximum, median, mean, variance, skewness, kurtosis*} of 25 MFCCs and {*mean and variance*} of the first and second MFCC derivatives (225-dimensional feature) achieved only 68% accuracy using the SVM classifier. This is worse than the performance of the best performing convnet feature.

It is notable again that unlike in the other tasks, concatenating convnet feature and MFCCs results in an improvement over either a convnet feature or MFCCs (71.4%). This suggests that they are complementary to each other in this task.

5. CONCLUSIONS

We proposed a transfer learning approach using deep learning and evaluated it on six music information retrieval and audio-related tasks. The pre-trained convnet was first trained to predict music tags and then aggregated features from the layers were transferred to solve genre classification, vocal/non-vocal classification, emotion prediction, speech/music classification, and acoustic event classification problems. Unlike the common approach in transfer learning, we proposed to use the features from every convolutional layers after applying an average-pooling to reduce their feature map sizes.

In the experiments, the pre-trained convnet feature showed good performance overall. It outperformed the baseline MFCC feature for all the six tasks, a feature that is very popular in music information retrieval tasks because it gives reasonable baseline performance in many tasks. It also outperformed the random-weights convnet features for all the six tasks, demonstrating the improvement by pre-training on a source task. Somewhat surprisingly, the performance of the convnet feature is also very competitive with state-of-the-art methods designed specifically for each task. The most important layer turns out to differ from task to task, but concatenating features from all the layers generally worked well. For all the five *music* tasks, concatenating MFCC feature onto convnet features did not improve the performance, indicating the music information in MFCC feature is already included in the convnet feature. We believe that transfer learning can alleviate the data sparsity problem in MIR and can be used for a large number of different tasks.

[6] Transfer learning targeting audio event classification was recently introduced in [2, 3] and achieved a state-of-the-art performance.

6. REFERENCES

[1] Arash Foroughmand Arabi and Guojun Lu. Enhanced polyphonic music genre classification using high level features. In *Signal and Image Processing Applications (ICSIPA), 2009 IEEE International Conference on*, pages 101–106. IEEE, 2009.

[2] Relja Arandjelović and Andrew Zisserman. Look, listen and learn. *arXiv preprint arXiv:1705.08168*, 2017.

[3] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, pages 892–900, 2016.

[4] Babu Kaji Baniya, Joonwhoan Lee, and Ze-Nian Li. Audio feature reduction and analysis for automatic music genre classification. In *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pages 457–462. IEEE, 2014.

[5] Thierry Bertin-Mahieux, Daniel PW Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 12th International Society for Music Information Retrieval Conference, October 24-28, 2011, Miami, Florida*, pages 591–596. University of Miami, 2011.

[6] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

[7] Keunwoo Choi, György Fazekas, Kyunghyun Cho, and Mark Sandler. The effects of noisy labels on deep convolutional neural networks for music classification. *arXiv:1706.02361*, 2017.

[8] Keunwoo Choi, György Fazekas, and Mark Sandler. Automatic tagging using deep convolutional neural networks. In *The 17th International Society of Music Information Retrieval Conference, New York, USA*. International Society of Music Information Retrieval, 2016.

[9] Keunwoo Choi, György Fazekas, and Mark Sandler. Explaining deep convolutional neural networks on music classification. *arXiv preprint arXiv:1607.02444*, 2016.

[10] Keunwoo Choi, György Fazekas, and Mark Sandler. Towards playlist generation algorithms using rnns trained on within-track transitions. In *Workshop on Surprise, Opposition, and Obstruction in Adaptive and Personalized Systems (SOAP), Halifax, Canada, 2016*, 2016.

[11] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Convolutional recurrent neural networks for music classification. In *2017 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2017.

[12] Keunwoo Choi, Deokjin Joo, and Juho Kim. Kapre: On-gpu audio preprocessing layers for a quick implementation of deep neural network models with keras. In *Machine Learning for Music Discovery Workshop at 34th International Conference on Machine Learning*. ICML, 2017.

[13] Keunwoo Choi, Jeonghee Kim, György Fazekas, and Mark Sandler. Auralisation of deep convolutional neural networks: Listening to learned features. In *International Society of Music Information Retrieval (ISMIR), Late-Breaking/Demo Session, Malaga, Spain*. International Society of Music Information Retrieval, 2015.

[14] François Chollet. Keras: Deep learning library for theano and tensorflow. https://github.com/fchollet/keras, 2015.

[15] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[16] Adam Coates and Andrew Y Ng. Learning feature representations with k-means. In *Neural Networks: Tricks of the Trade*, pages 561–580. Springer, 2012.

[17] Aggelos Gkiokas, Vassilis Katsouros, and György Carayannis. Towards multi-purpose spectral rhythm features: An application to dance style, meter and tempo estimation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(11):1885–1896, 2016.

[18] Fabien Gouyon, Simon Dixon, Elias Pampalk, and Gerhard Widmer. Evaluating rhythmic descriptors for musical genre classification. In *25th AES International Conference, London, UK*, 2004.

[19] Grzegorz Gwardys and Daniel Grzywczak. Deep image features in music information retrieval. *International Journal of Electronics and Telecommunications*, 60(4):321–326, 2014.

[20] Philippe Hamel, Matthew EP Davies, Kazuyoshi Yoshii, and Masataka Goto. Transfer learning in mir: Sharing learned latent representations for music audio classification and similarity. Curitiba, Brazil, 2013.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[22] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: a new learning scheme of feedforward neural networks. In *IEEE International Joint Conference on Neural Networks*, volume 2, pages 985–990. IEEE, 2004.

[23] Yin-Fu Huang, Sheng-Min Lin, Huan-Yu Wu, and Yu-Siou Li. Music genre classification based on local feature selection using a self-adaptive harmony search algorithm. *Data & Knowledge Engineering*, 92:60–76, 2014.

[24] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[25] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[27] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, volume 14, pages 1188–1196, 2014.

[28] Jongpil Lee and Juhan Nam. Multi-level and multi-scale feature aggregation using pre-trained convolutional neural networks for music auto-tagging. *arXiv preprint arXiv:1703.01793*, 2017.

[29] Dawen Liang, Minshu Zhan, and Daniel PW Ellis. Content-aware collaborative music recommendation using pre-trained neural networks. In *Conference of the International Society for Music Information Retrieval (ISMIR 2015)*, pages 295–301. Malaga, Spain, 2015.

[30] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[31] Athanasios Lykartsis and Alexander Lerch. Beat histogram features for rhythm-based musical genre classification using multiple novelty functions. In *Proceedings of the 16th ISMIR Conference*, pages 434–440, 2015.

[32] Ugo Marchand and Geoffroy Peeters. The extended ballroom dataset. *Conference of the International Society for Music Information Retrieval (ISMIR 2016) late-breaking session*, 2016.

[33] Ugo Marchand and Geoffroy Peeters. Scale and shift invariant time/frequency representation using auditory statistics: Application to rhythm description. In *Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on*, pages 1–6. IEEE, 2016.

[34] Brian McFee, Matt McVicar, Colin Raffel, Dawen Liang, Oriol Nieto, Eric Battenberg, Josh Moore, Dan Ellis, Ryuichi YAMAMOTO, Rachel Bittner, Douglas Repetto, Petr Viktorin, João Felipe Santos, and Adrian Holovaty. librosa: 0.4.1, October 2015.

[35] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[36] Brian CJ Moore. *An introduction to the psychology of hearing*. Brill, 2012.

[37] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724, 2014.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[39] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.

[40] Jordi Pons and Xavier Serra. Designing efficient architectures for modeling temporal features with convolutional neural networks. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2017.

[41] Mathieu Ramona, Gaël Richard, and Bertrand David. Vocal detection in music with support vector machines. In *Acoustics, Speech and Signal Processing (ICASSP), 2008 IEEE International Conference on*, pages 1885–1888, March 31 - April 4 2008.

[42] J. Salamon, C. Jacoby, and J. P. Bello. A dataset and taxonomy for urban sound research. In *22st ACM International Conference on Multimedia (ACM-MM'14)*, Orlando, FL, USA, Nov. 2014.

[43] Justin Salamon and Juan Pablo Bello. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 2017.

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[45] Alex J Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.

[46] Mohammad Soleymani, Micheal N Caro, Erik M Schmidt, Cheng-Ya Sha, and Yi-Hsuan Yang. 1000 songs for emotional analysis of music. In *Proceedings of the 2nd ACM international workshop on Crowdsourcing for multimedia*, pages 1–6. ACM, 2013.

[47] M Srinivas, Debaditya Roy, and C Krishna Mohan. Learning sparse dictionaries for music and speech classification. In *Digital Signal Processing (DSP), 2014 19th International Conference on*, pages 673–675. IEEE, 2014.

[48] Bob L Sturm. The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use. *arXiv preprint arXiv:1306.1461*, 2013.

[49] Bob L Sturm et al. Revisiting priorities: Improving mir evaluation practices. In *Proc. 17th International Society for Music Information Retrieval Conference (ISMIR'16), New York, NY, USA*, 2016.

[50] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.

[51] The Theano Development Team, Rami Al-Rfou, Guillaume Alain, Amjad Almahairi, Christof Angermueller, Dzmitry Bahdanau, Nicolas Ballas, Frédéric Bastien, Justin Bayer, Anatoly Belikov, et al. Theano: A python framework for fast computation of mathematical expressions. *arXiv preprint arXiv:1605.02688*, 2016.

[52] George Tzanetakis. Gtzan musicspeech. *availabe online at http://marsyas.info/download/data sets*, 1999.

[53] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *Speech and Audio Processing, IEEE transactions on*, 10(5):293–302, 2002.

[54] Aäron Van Den Oord, Sander Dieleman, and Benjamin Schrauwen. Transfer learning by supervised pretraining for audio-based music classification. In *Conference of the International Society for Music Information Retrieval (ISMIR 2014)*, 2014.

[55] Chia-Ming Wang and Yin-Fu Huang. Self-adaptive harmony search algorithm for optimization. *Expert Systems with Applications*, 37(4):2826–2837, 2010.

[56] Felix Weninger, Florian Eyben, and Bjorn Schuller. On-line continuous-time music mood regression with deep recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 5412–5416. IEEE, 2014.

[57] Jason Weston, Samy Bengio, and Nicolas Usunier. Wsabie: Scaling up to large vocabulary image annotation. *International Joint Conference on Artificial Intelligence*, 2011.

[58] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[59] Yujun Zeng, Xin Xu, Yuqiang Fang, and Kun Zhao. Traffic sign recognition using extreme learning classifier with deep convolutional features. In *The 2015 international conference on intelligence science and big data engineering (IScIDE 2015), Suzhou, China*, 2015.