Multitask Learning based Deep Learning Model for Music Artist and Language Recognition

Yeshwant Singh **Anupam Biswas**
Department of Computer Science and Engineering,
National Institute of Technology Silchar, Assam, India, 788010
{yeshwant_rs, anupam}@cse.nits.ac.in

Abstract

Artist and music language recognitions of music recordings are crucial tasks in the music information retrieval domain. These tasks have many industrial applications and become much important with the advent of music streaming platforms. This work proposed a multitask learning-based deep learning model that leverages the shared latent representation between these two related tasks. Experimentally, we observe that applying multitask learning over a simple few blocks of a convolutional neural network-based model pays off with improvement in the performance. We conduct experiments on a regional music dataset curated for this task and released for others. Results show improvement up to 8.7 percent in AUC-PR, similar improvements observed in AUC-ROC.

1 Introduction

Music is a universal language that we innately understand. It can influence or induce new emotions in the listeners. Artists project their emotions and feeling onto their music that is felt and observed in the music. Artist recognition of a music recording is an active area of research in music information retrieval (MIR) (Mesaros et al., 2007; Sharma et al., 2019; Hu et al., 2021) and has various applications.

Artist recognition is crucial in the areas of music index, retrieval, and recommendation. The digitization of the music industry and music streaming platforms have created large volumes of digital music that need to be processed and stored on a large scale. This has reignited the research in the music domain. Recognition of the artist of a song is crucial for these music streaming platforms. We also have our favorite artists, whom we search for on these streaming platforms as music listeners, and it shows how vital artist recognition is.

Machine learning-based approaches treat this problem as a multi-label classification problem. There have been many recent deep learning-based techniques that perform very well for this task. The approach in these techniques is to use variants of spectrogram and train the deep neural network model on that visual representation (Yu and Slotine, 2009; Kalantarian et al., 2014; Wu et al., 2018). Some techniques have used raw waveforms to train sequence-based models. These techniques have revealed that related tasks specific noise filtering can boost the overall generalization of deep learning-based models for music-related tasks.

Surprisingly, given techniques by researchers do not leverage the shared representation learned by multitask learning of related tasks. In this paper, we propose a multitask learning-based model for artist recognition that leverages the shared representation learned from the related task of music language recognition. The results show improvement over single-task learning. We have used multitask learning with convolutional neural networks (CNN) for artist recognition, and we observed improved performance.

Multitask learning is a machine learning paradigm in which related tasks are trained together using the same model which shares bottom layers (in neural networks) among the related tasks. The training signals (gradients) from different related tasks force the model to learn more generalized data representation by filtering out noise for each related task (Böck et al., 2019; Zeng et al., 2019). Alternatively, we can say the knowledge learned for a task helps in the performance over another related task. In our case, we use two related tasks of artist recognition and music language recognition, where artist recognition is the primary task for leveraging shared representation from multitask learning. Music recordings spectrograms are used as an input for the model, and corresponding artists and language are predicted as an output by the model.

This paper is organized as follows: Section 2

Proceedings of the Workshop on Speech and Music Processing (SMP-2021), pages 20–23
co-located with ICON-2021, December 16-19, 2021, Silchar, India.

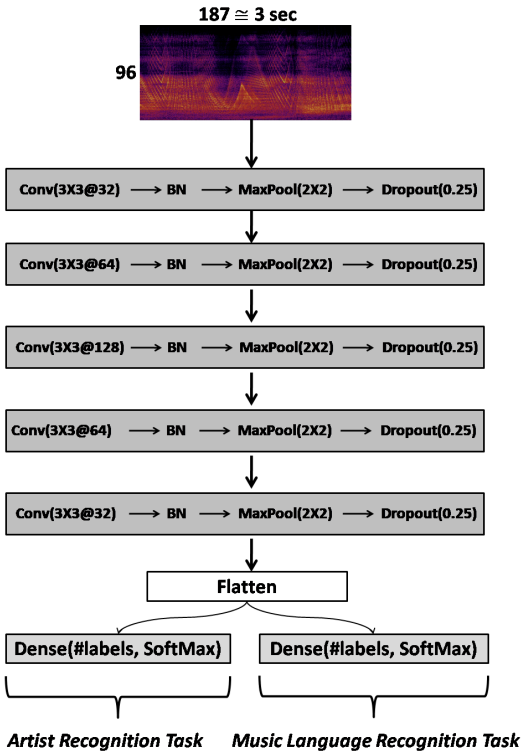

Figure 1: Proposed Multitask learning based model architecture.

presents the proposed approach. The dataset, experimental setup is summarized in Section 3. In Section 4, we present the results and discussions of our findings. Finally, we conclude in Section 5.

2 Proposed Method

Our proposed approach is a model based on convolutional neural networks (CNN) similar to VGG architecture. CNNs are very popular in the computer vision domain. Here we have chosen CNN because the spectrogram representation of music samples can be treated as an image, and CNNs can extract relevant knowledge from them. This spectrogram approach is not new and has been used in the past with traditional machine learning. However, advances in computer vision have given new avenues in the spectrogram representation for music tasks.

The architecture of our model is shown in Figure 1. A mel-spectrogram taken as a 2D-tensor with (187x96) is taken as an input to the network. The mel-spectrogram consists of 187 frames which correspond to 3 seconds of music data and 96 mel-bands. The extraction of the spectrogram from the music recordings is discussed in Section 3.2.1.

A batch of spectrograms with size 64 is passed to five blocks of CNN layers. Each block consists of a CNN layer along with batch normalization, max-pooling, and dropout. The kernel size of CNN layers is fixed to (3x3), and the numbers of channels are 32, 64, 128, 64, and 32, respectively, with stride 1 of CNN layers in five blocks. The non-linearity in the CNN is set ReLU. After each CNN layer, batch normalization (BN) is applied to normalize the weights during training. Following BN, max-pooling is applied with a pool size of (2x) and stride of (2x2). Lastly, a dropout layer with a 0.25 drop rate is applied.

After these CNN layer blocks, a flatten layer is used to compress the output shape from CNN blocks to the 1D tensor. Dense layers then use this tensor corresponding to each related task (two in our case, namely artist recognition, and music language recognition). Each dense layer has the number of units equal to the number of labels in the corresponding task (64, 17 in our tasks). The softmax activation function is used in these dense layers' outputs to get the probability distribution of labels relating to the tasks.

These are 193k trainable parameters in the top five blocks of our proposed model. The parameters in the output layers are dependent on the number of parameters in the previous layer and the number of labels in the corresponding task. The model shares the five blocks of the model between two related tasks. The representation learning during the training of these two tasks forces the model to generalize better than it would have been training on a single task. The loss used for training the model is a weighted sum of losses from individual tasks. Categorical cross-entropy is used as a loss for both tasks. The weighted sum for overall loss function:

$$L = L_{artist} + \alpha L_{lang}$$

Here, L_{artist} and L_{lang} denote the corresponding loss for artist and music language recognition tasks, respectively. α represents the hyper-parameter for the weightage given to the loss of music language recognition task in the overall loss. Setting it to zero disables the multitask learning and is done when we are done with training the network. The choice of α is discussed in Section 3.2.4.

3 Experiments

3.1 Dataset Description

The model is trained on a dataset prepared for regional music by us (Singh, 2021). The dataset consists of 17 languages: Hindi, Gujarati, Marathi, Konkani, Bengali, Oriya, Kashmiri, Assamese, Nepali, Konyak, Manipuri, Khasi & Jaintia, Tamil, Malayalam, Punjabi, Telugu, Kannada.

For each language, four artists are chosen (two male and two female), and for each artist, five songs are collected. The artists are chosen considering the veteran and contemporary artists. So, two out of four artists are veteran performers, and the remaining two are modern artists. Overall, there are 68 artists and 340 music songs with 23.2 hours of duration.

3.2 Experimental Setup

3.2.1 Preprocessing

A preprocessing step is performed over the music recordings before feeding them to the model. Music recordings are converted to mel-spectrograms of 3-sec segments. These 3-sec segments are created from two 1 minute long pieces taken out from each music recording. The spectrograms are generated from resampled waveforms with 96 mel-bands. The shape of the spectrogram is $(t \times 96)$, where t is the number of frames (proportional to time). Librosa library (McFee et al., 2015) is used for the extraction of mel-spectrogram from 3-sec music segments.

3.2.2 Evaluation Metrics

We have used average precision (AP) as an evaluation metric. It is commonly used in multi-label classification tasks. It is the weighted average of precision values across different recall values, or it can be said as the area under the precision-recall curve (AUC-PR). We also report another evaluation metric called the area of the receiver operating characteristic curve (AUC-ROC).

3.2.3 Training

The training of the model is performed with a batch size of 32. As we take 3-sec long music data for generating spectrogram, they must be batched together to speed up the training process. Tensorflow is used as a deep learning framework for building and training our model. This framework handles the batching of spectrograms.

We split the data into three splits of 80, 10, and 10 percent of samples for training, validation, and testing purposes. The splits are ensured to happen at song level instead of segment level to ensure that segments from a few songs are not just present in validation and testing datasets, preventing data leakage. Adam optimizer is used for training the model with a 0.001 learning rate.

3.2.4 Multitask Learning Weighted Loss Function

The overall loss function is defined as a weighted sum of the losses of individual tasks. That is $L = L_{artist} + \alpha L_{lang}$. The hyperparameter α influences the contribution of L_{lang} in the overall loss. We tried different values for α for training the model on the mentioned dataset to get the optimal value. We found following formula works well in our scenario:

$$\alpha = \frac{N_{artist}}{N_{lang}}$$

Here, N_{artist} and N_{lang} are the number of labels in the artist and music language recognition tasks. It can be said that α balances the numbers of labels in the given tasks. The α for our experiments is computed using the above formula.

4 Results

We performed multiple experiments over the described dataset in Section 3.1. Baseline and Multitask are two models selected and trained for two tasks. Both models are the same architecture except for the final dense layer. The Baseline model has a single dense layer having units equal to the labels in the given task. While in the multitask model, there are two dense layers side by side for each task, having units equal to two task labels. Comparing the result of baseline and multitask models allow us to observe the impact of multitask learning on the given task with the help of additional related task. The comparative analysis is presented in Table 1, which shows models performance across different configurations.

We report that the multitask model shows improvement over the baseline model for both tasks and can be observed in both evaluation metrics. On the curated dataset, we observe an increase of 4.43 percent for artist recognition in the AUC-ROC metric, while the AUC-PR metric recorded a 7.48 improvement. For the music language recognition task, improvements are even further. 5.55 improvement is observed in the AUC-ROC and 8.69 for

Model type	N_{artist}	N_{lang}	# songs	# 1-min segments	# 3-sec segments	AUC-ROC (%)	AUC-PR (%)
Baseline	68	17	340	680	1360	70.45	39.78
Multi-task I	68	17	340	680	1360	74.88 (**+4.43**)	47.26 (**+7.48**)
Multi-task II	68	17	340	680	1360	76.00 (**+5.55**)	48.47 (**+8.69**)

Table 1: Baseline and Multitask models performance reported in AUC-ROC and AUC-PR metrics. N_{artist} and N_{lang} represents the number of labels in artist recognition and music language recognition tasks, repectively. **(+x.xx)** represents the increase in the given evaluation metric compared to the performance of Baseline model.

the AUC-PR metrics. We observe from these experiments that the performance can be improved by adding more related tasks and increasing the number of data samples.

5 Conclusions

In this paper, we present the importance of artist recognition from the perspective of music streaming platforms for storage, indexing and music information retrieval tasks. It is crucial to building a more generalized system by these platforms.

Towards building a more generalized system, we observe that multitask learning can help achieve a more generalized system by leveraging the model's representation across different related tasks. We propose a multitask learning model for artist and music language recognition tasks. Experiments depict that the multitask learning approach improves the performance of the single-task baseline model.

Acknowledgement

This research was funded under the grant number: ECR/2018/000204 by the Science & Engineering Research Board (SERB).

References

Sebastian Böck, Matthew EP Davies, and Peter Knees. 2019. Multi-task learning of tempo and beat: Learning one to improve the other. In *ISMIR*, pages 486–493.

Shichao Hu, Beici Liang, Zhouxuan Chen, Xiao Lu, Ethan Zhao, and Simon Lui. 2021. Large-scale singer recognition using deep metric learning: an experimental study. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE.

Haik Kalantarian, Nabil Alshurafa, Mohammad Pourhomayoun, Shruti Sarin, Tuan Le, and Majid Sarrafzadeh. 2014. Spectrogram-based audio classification of nutrition intake. In *2014 IEEE Healthcare Innovation Conference (HIC)*, pages 161–164. IEEE.

Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, pages 18–25. Citeseer.

Annamaria Mesaros, Tuomas Virtanen, and Anssi Klapuri. 2007. Singer identification in polyphonic music using vocal separation and pattern recognition methods. In *ISMIR*, pages 375–378.

Bidisha Sharma, Rohan Kumar Das, and Haizhou Li. 2019. On the importance of audio-source separation for singer identification in polyphonic music. In *INTERSPEECH*, pages 2020–2024.

Yeshwant Singh. 2021. Regional music dataset. https://github.com/yeshwantsingh/regional_dataset. (Accessed on 11/14/2021).

Yu Wu, Hua Mao, and Zhang Yi. 2018. Audio classification using attention-augmented convolutional neural network. *Knowledge-Based Systems*, 161:90–100.

Guoshen Yu and Jean-Jacques Slotine. 2009. Audio classification from time-frequency texture. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1677–1680. IEEE.

Yuni Zeng, Hua Mao, Dezhong Peng, and Zhang Yi. 2019. Spectrogram based multi-task audio classification. *Multimedia Tools and Applications*, 78(3):3705–3722.