SAMPLE-LEVEL CNN ARCHITECTURES FOR MUSIC AUTO-TAGGING USING RAW WAVEFORMS

Taejun Kim[1], Jongpil Lee[2], Juhan Nam[2]

[1] School of Electrical and Computer Engineering, University of Seoul,
[2] Graduate School of Culture Technology, KAIST,
ktj7147@uos.ac.kr, {richter, juhannam}@kaist.ac.kr

ABSTRACT

Recent work has shown that the end-to-end approach using convolutional neural network (CNN) is effective in various types of machine learning tasks. For audio signals, the approach takes raw waveforms as input using an 1-D convolution layer. In this paper, we improve the 1-D CNN architecture for music auto-tagging by adopting building blocks from state-of-the-art image classification models, *ResNets* and *SENets*, and adding multi-level feature aggregation to it. We compare different combinations of the modules in building CNN architectures. The results show that they achieve significant improvements over previous state-of-the-art models on the MagnaTagATune dataset and comparable results on Million Song Dataset. Furthermore, we analyze and visualize our model to show how the 1-D CNN operates.

Index Terms— convolutional neural networks, music auto-tagging, raw waveforms, multi-level learning

1. INTRODUCTION

Time-frequency representations based on short-time Fourier transform, often scaled in a log-like frequency such as mel-spectrogram, are the most common choice of input in the majority of state-of-the-art music classification algorithms [1, 2, 3, 4, 5]. The 2-dimensional input represents acoustically meaningful patterns well but requires a set of parameters, such as window size/type and hop size, which may have different optimal settings depending on the type of input signals.

In order to overcome the problem, there have been some efforts to directly use raw waveforms as input particularly for convolutional neural networks (CNN) based models [6, 7]. While they show promising results, the models used large filters, expecting them to replace the Fourier transform. Recently, Lee et. al. [8] addressed the problem using very small filters and successfully applied the 1D CNN to the music auto-tagging task. Inspired from the well-known *VGG* net that uses very small size of filters such as 3×3, [9], the sample-level CNN model was configured to take raw waveforms as input and have filters with such small granularity.

A number of techniques to further improve performances of CNNs have appeared recently in image domain. He et. al. introduced *ResNets* which includes *skip connections* that enables a very deep CNN to be effectively trained and makes gradient propagation fluent [10]. Using the skip connections, they could successfully train a 1001-layer ResNet [11]. Hu et. al proposed *SENets* [12] which includes a building block called *Squeeze-and-Excitation* (SE). Unlike other recent approaches, the block concentrates on channel-wise information, not spatial. The SE block adaptively recalibrates feature maps using a channel-wise operation. Most of the techniques were developed in the field of computer vision but they are not fully adopted for music classification tasks. Although there were a few approaches to readily apply them to audio domain [7, 13]. They used 2D representations as input [13] or used large filters for the first 1D convolutional layer [7].

On the other hand, some methods are concerned with overall architecture of the model rather than designing a fine-grained building block [2, 14, 15, 16, 17]. Specifically, multi-level feature aggregation combines several hidden layer representations for final prediction [2, 14]. They significantly improved the performance in music auto-tagging by taking different levels of abstractions of tag labels into account.

In this paper, we explore the building blocks of advanced CNN architectures, *ResNets* and *SENets*, based on the sample-level CNN for music auto-tagging. Also, we observe how the multi-level feature aggregation affects the performance. The results show that they achieve significant improvements over previous state-of-the-art models on the MagnaTagATune dataset and comparable results on Million Song Dataset. Furthermore, we analyze and visualize our model built with the SE blocks to show how the 1D CNN operates. The results show that the input signals are processed in a different manner depending on the level of layers.

2. ARCHITECTURES

All of our models are based on the sample-level 1D CNN model [8], which is constructed with the basic block shown in Figure 1(b). Every filter size of the convolution layers is fixed

(a) Overview of the architecture   (b) Basic block [8]   (c) SE block   (d) Res-n block   (e) ReSE-n block
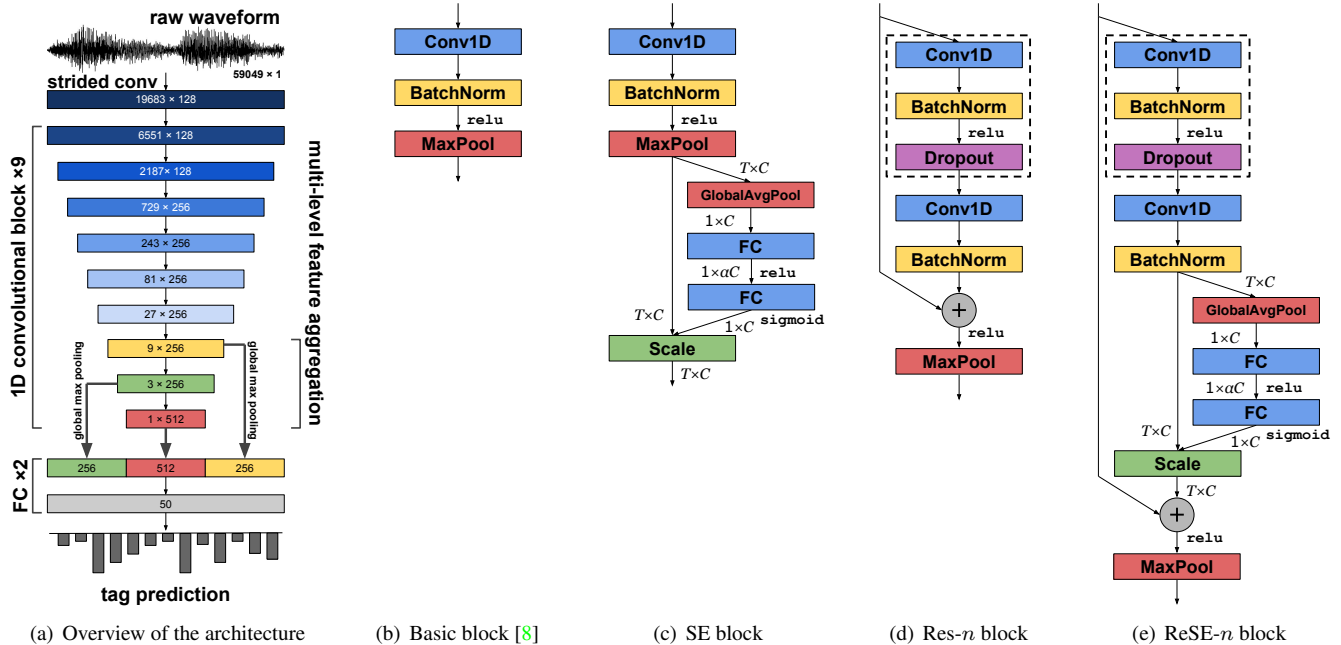
Fig. 1. The proposed architecture for music auto-tagging. (a) The models consist of a strided convolutional layer, 9 blocks, and two fully-connected (FC) layers. The outputs of the last three blocks are concatenated and then used as input of the last two FC layers. Output dimensions of each block (or layer) are denoted inside of them (temporal×channel). (b-e) The 1D convolutional building blocks that we evaluate.

to three. The differences between the sample-level CNN and ours are the use of advanced building blocks and multi-level feature aggregation. In this section, we describe the details.

2.1. 1D convolutional building blocks

2.1.1. SE block

We utilize the *SE* block from *SENets* to increase representational power of the basic block. As shown in Figure 1(c), we simply attached the SE block to the basic block. The SE block recalibrates feature maps from the basic block through two operations. One is *squeeze* operation that aggregates a global temporal information into channel-wise statistics using global average pooling. The operation reduces the temporal dimensionality (T) to one, averaging outputs from each channel. The other is *excitation* operation that adaptively recalibrates feature maps of each channel using the channel-wise statistics from the squeeze operation and a simple gating mechanism. The gating mechanism consists of two fully-connected (FC) layers that compute nonlinear interactions among channels. Finally, the original outputs from the basic block are rescaled by channel-wise multiplication between the feature map and the sigmoid activation of the second FC layer.

Unlike the original SE block in *SENets*, our excitation operation does not form a bottleneck. On the contrary, we expand the channel dimensionality (C) to αC at the first FC

layer, and then reduce the dimensionality back to C at the second layer. We set the amplifying ratio α to be 16, after a grid search with $\alpha = [2^{-3}, 2^{-2}, ..., 2^6]$.

2.1.2. Res-n block

Inspired by *skip connections* from *ResNets*, we modified the basic block by adding a skip connection as shown in Figure 1(d). *Res-n* denotes that the block uses n convolutional layers where n is one or two. Specifically, *Res-2* is a block that has the additional layers denoted by the dotted line in Figure 1(d), and *Res-1* is a block that has a skip connection only. When the block uses two convolutional layers (*Res-2*), we add a dropout layer (with a drop ratio of 0.2) between two convolutions to avoid overfitting. This technique was firstly introduced at *WideResNets* [18].

2.1.3. ReSE-n block

The *ReSE-n* block is a combination of the SE and Res-n blocks as shown in Figure 1(e). n denotes the number of convolutional layers in the block, where n is also one or two. A dropout layer is inserted when n is two.

2.2. Multi-level feature aggregation

Fig. 1(a) shows the multi-level feature aggregations that we configured. The outputs of the last three blocks are concate-

Table 1. AUCs of CNN architectures on MTAT. "multi" and "no multi" indicates if the multi-level feature aggregation is used or not. † denotes using a weight decay of 10^{-4}.

Block	MTAT	
	multi	no multi
Basic [8]	0.9077	0.9055
SE	**0.9111**	0.9083
Res-1	0.9037	0.9048
Res-2	0.9098	0.9061
ReSE-1	0.9053	0.9066
ReSE-2	**0.9113**†	0.9102†

Table 2. AUCs of state-of-the-art models on MTAT and MSD. † denotes that the model used an ensemble of three.

Model	MTAT	MSD
Bag of multi-scaled features [3]	0.8980	-
End-to-end [6]	0.8815	-
Transfer learning [4]	0.8800	-
Persistent CNN [21]	0.9013	-
Time-Frequency CNN [22]	0.9007	-
Timbre CNN [23]	0.8930	-
2D CNN [5]	0.8940	0.8510
CRNN [1]	-	0.8620
Multi-level & multi-scale [2]	0.9017†	**0.8878**†
SampleCNN multi-features [14]	0.9064†	0.8842
SampleCNN [8]	0.9055	0.8812
SE [This work]	**0.9111**	0.8840
ReSE [This work]	**0.9113**	0.8847

nated and then delivered to the FC layers. Before the concatenation, temporal dimensions of the outputs are reduced to one by a global max pooling. Unlike [2], the concatenation occurs while training the CNN and the average pooling over the whole audio clip (i.e. 29 second long), which followed by the global max pooling, is not included.

3. EXPERIMENTS

3.1. Datasets

We evaluated the proposed architectures on two datasets, MagnaTagATune (MTAT) dataset [19] and Million Song Dataset (MSD) annotated with the Last.FM tags [20]. We split and filtered both of the datasets, following the previous work [5, 6, 8]. We used the 50 most frequent tags. All songs are trimmed to 29 seconds long, and resampled to 22050Hz as needed. The song is divided into 10 segments of 59049 samples. To evaluate the performance of music auto-tagging which is a multi-class and multi-label classification task, we computed the Area Under the Receiver Operating Characteristic curve (AUC) for each tag and computed the average across all 50 tags. During the evaluation, we average predictions across all segments.

3.2. Implementation details

All the networks were trained using SGD with Nesterov momentum of 0.9 and mini-batch size 23. The initial learning rate is set to 0.01, decayed by a factor of 5 when a validation loss is on a plateau. None of the regularizations are used on MSD. A dropout layer of 0.5 was inserted before the last FC layer on MTAT. For all building blocks, we evaluated either with or without the multi-level feature aggregation. Since the training for MSD takes much time longer than MTAT, we explored the architectures mainly on MTAT, and then trained the two best models on MSD. Code and models built with TensorFlow and Keras are available at the link[1].

---

[1] https://github.com/tae-jun/resemul

4. RESULTS AND DISCUSSION

### 4.1. Comparison of the architectures

Table 1 summarizes the evaluation results of compared CNN architectures on the MTAT dataset. They show that the SE block is more effective than the Res-n blocks, increasing the performance of the basic block for all cases. In the Res-n block, only adding the skip connection to the basic block (Res-1) actually decreases the performance. The combination of the SE and the Res-2 improves it slightly more. However, a training time of the ReSE-2 is 1.8 times longer than the basic block whereas the SE block only 1.08 times longer. Thus, if the training or prediction time of the models is important, the SE model will be preferred to the ReSE-2. The effect of the multi-level aggregation is valid for the majority of the models. We obtained two best results in Table 1 by using the multi-level aggregation.

4.2. Comparison with state-of-the-arts

Table 2 compares previous state-of-the-art models in music auto-tagging with our best models, the SE block and ReSE-2 block, each with multi-level aggregation. On the MTAT dataset, our best models outperform all the previous results. On MSD, they are not the best but are comparable to the second-tier.

5. ANALYSIS OF EXCITATION

To lay the groundwork for understanding how 1D CNNs operate, we analyze the sigmoid activations of *excitations* in the SE blocks at different levels graphically and quantitatively. In this section, we observe how the SE blocks recalibrate channels, depending on which level they exist. The blocks used for the analysis are from the SE model using the multi-level
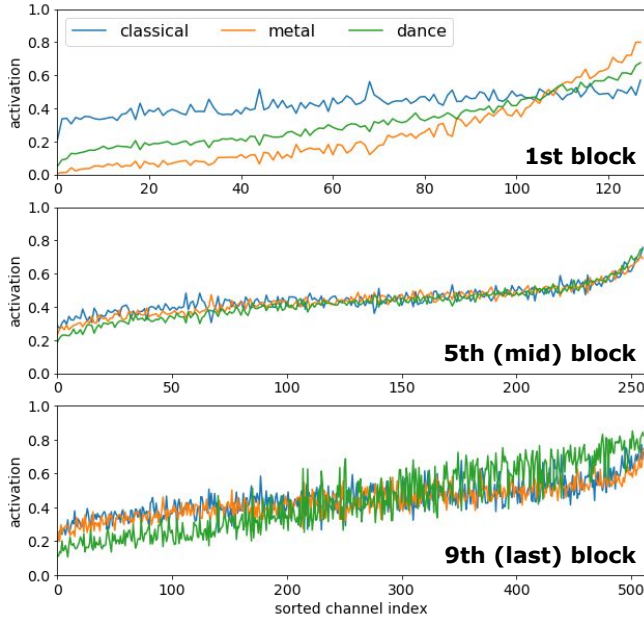
Fig. 2. Visualization of the sigmoid activations of *excitations* in the *SE* model. The channel index was sorted by the average of the activations.

Table 3. Co-occurrence matrix of the tags used in Figure 2

	classical	metal	dance
classical	704	**0**	**1**
metal	**0**	166	**0**
dance	**1**	**0**	153

Fig. 3. Standard deviations (std) of the activations of *excitations* across all tags along each layer.

feature aggregation and they were trained on MTAT. The activations were extracted from its test set. The activations were averaged over all segments separately for each tag.
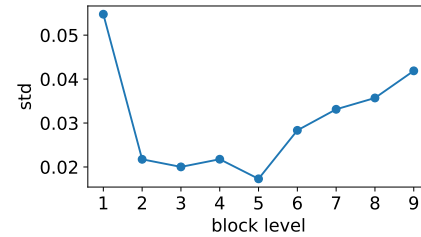
5.1. Graphical analysis

For this analysis, we chose three tags, *classical*, *metal*, and *dance* that are not similar to each other as shown in Table 3. Figure 2 shows the average sigmoid activations in the SE blocks for the songs with the three tags. The different levels of activations indicate that the SE blocks process input audio differently depending on the tag (or genre) of the music. That is, every block in Figure 2 fires different patterns of activations for each tag at a specific channel. This trend is strongest at the first block (top), weakest at the mid block (middle), and becomes stronger again at the last block (bottom).

This trend is somewhat different from what are observed in the image domain [12], where the exclusiveness of average excitation for input with different labels are monotonically increasing along the layers. Specifically, the first block fires high activations for *classical*, low ones for *dance*, and even lower ones for *metal* for the majority of the channels. On the other hand, the activations of the last block vary depending on the tags. For example, the activations of *metal* are high at some channels but low at the others, which makes the activations noisy even though they are sorted. We can interpret this result as follows. The first block normalizes the loudness of the audios because the block fires high activations for *classical* music, which tend to have small volume, and low activa-

tions for *metal* music, which tend to have large volume. Also, the middle block processes common features among them as they have similar levels of activations. Finally, the noisy exclusiveness in the last block indicates that they effectively discriminate the music with different tags.

5.2. Quantitative analysis

We assure the exclusiveness trend by measuring standard deviations of the activations across all tags at every level. Figure 3 shows that the higher the standard deviation is, the more the block responses to the song differently according to its tag. The result shows that the standard deviation is highest at the first block, it drops and stays low up to the 5th block and then increases gradually until the last block. That is, the four lower blocks except the the bottom one (2 to 5) tend to handle general features whereas the four upper blocks (6 to 9) tend to progressively more discriminative features.

6. CONCLUSION

We proposed 1D convolutional building blocks based on the previous work, the *sample-level CNN*, *ResNets*, and *SENets*. The *ReSE* block, which is a combination of the three models, showed the best performance. Also, the multi-level feature aggregation showed improvements on the majority of the building blocks. Through the experiments, we obtained state-of-the-art performance on the MTAT dataset and high-ranked results on MSD. In addition, we analyzed the activations of *excitation* in *SE* model to understand the effect. With this analysis, we could observe that the SE blocks process non-similar songs exclusively and how the different levels of the model process the songs in a different manner.

7. REFERENCES

[1] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho, "Convolutional recurrent neural networks for music classification," in *ICASSP*. IEEE, 2017, pp. 2392–2396.

[2] Jongpil Lee and Juhan Nam, "Multi-level and multi-scale feature aggregation using pretrained convolutional neural networks for music auto-tagging," *IEEE Signal Processing Letters*, vol. 24, no. 8, pp. 1208–1212, 2017.

[3] Sander Dieleman and Benjamin Schrauwen, "Multi-scale approaches to music audio feature learning," in *International Society of Music Information Retrieval Conference (ISMIR)*, 2013, pp. 116–121.

[4] Aäron Van Den Oord, Sander Dieleman, and Benjamin Schrauwen, "Transfer learning by supervised pre-training for audio-based music classification," in *International Society of Music Information Retrieval Conference (ISMIR)*, 2014.

[5] Keunwoo Choi, György Fazekas, and Mark B. Sandler, "Automatic tagging using deep convolutional neural networks," in *International Society of Music Information Retrieval Conference (ISMIR)*, 2016.

[6] Sander Dieleman and Benjamin Schrauwen, "End-to-end learning for music audio," in *ICASSP*. IEEE, 2014, pp. 6964–6968.

[7] Wei Dai, Chia Dai, Shuhui Qu, Juncheng Li, and Samarjit Das, "Very deep convolutional neural networks for raw waveforms," in *ICASSP*. IEEE, 2017, pp. 421–425.

[8] Jongpil Lee, Jiyoung Park, Keunhyoung Luke Kim, and Juhan Nam, "Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms," in *Sound and Music Computing Conference (SMC)*, 2017.

[9] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 630–645.

[12] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-excitation networks," *arXiv preprint arXiv:1709.01507*, 2017.

[13] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin W. Wilson, "Cnn architectures for large-scale audio classification," in *ICASSP*. IEEE, 2017, pp. 131–135.

[14] Jongpil Lee and Juhan Nam, "Multi-level and multi-scale feature aggregation using sample-level deep convolutional neural networks for music classification," *Machine Learning for Music Discovery Workshop, International Conference on Machine Learning (ICML)*, 2017.

[15] Yi Sun, Xiaogang Wang, and Xiaoou Tang, "Deep learning face representation from predicting 10,000 classes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1891–1898.

[16] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *International Conference on Machine Learning (ICML)*, 2014, pp. 647–655.

[17] Yusuf Aytar, Carl Vondrick, and Antonio Torralba, "Soundnet: Learning sound representations from unlabeled video," in *Neural Information Processing Systems (NIPS)*, 2016, pp. 892–900.

[18] Sergey Zagoruyko and Nikos Komodakis, "Wide residual networks," *British Machine Vision Conference (BMVC)*, 2016.

[19] Edith Law, Kris West, Michael I. Mandel, Mert Bay, and J. Stephen Downie, "Evaluation of algorithms using games: The case of music tagging," in *International Society of Music Information Retrieval Conference (ISMIR)*, 2009.

[20] Thierry Bertin-Mahieux, Daniel P. W. Ellis, Brian Whitman, and Paul Lamere, "The million song dataset," in *International Society of Music Information Retrieval Conference (ISMIR)*, 2011, vol. 2, p. 10.

[21] Jen-Yu Liu, Shyh-Kang Jeng, and Yi-Hsuan Yang, "Applying topological persistence in convolutional neural network for music audio signals," *arXiv preprint arXiv:1608.07373*, 2016.

[22] Umut Güçlü, Jordy Thielen, Michael Hanke, Marcel van Gerven, and Marcel AJ van Gerven, "Brains on beats," in *Neural Information Processing Systems (NIPS)*, 2016, pp. 2101–2109.

[23] Jordi Pons, Olga Slizovskaia, Rong Gong, Emilia Gómez, and Xavier Serra, "Timbre analysis of music audio signals with convolutional neural networks," *arXiv preprint arXiv:1703.06697*, 2017.