Knowledge Transfer from Neural Networks for Speech Music Classification

Christian Kehling[1,2] and Estefanía Cano[3] ⋆

[1] Institute for Digital Media Technology
Technical Univeristy of Ilmenau
[2] Fraunhofer Institute for Digital Media Technology
[3] Songquito UG
christian.kehling@tu-ilmenau.de

Abstract. A frequent problem when dealing with audio classification tasks is the scarcity of suitable training data. This work investigates ways of mitigating this problem by applying transfer learning techniques to neural network architectures for several classification tasks from the field of Music Information Retrieval (MIR). First, three state-of-the-art architectures are trained and evaluated with several datasets for the task of speech/music classification. Second, feature representations or embeddings are extracted from the trained networks to classify new tasks with unseen data. The effect of pre-training with respect to the similarity of the source and target tasks are investigated in the context of transfer learning, as well as different fine-tuning strategies.

Keywords: Deep Learning, Neural Networks, Audio Classification, Speech Music Classification, Transfer Learning, Embeddings, Music Information Retrieval

1 Introduction

Detection of speech and music in audio signals has been investigated in the field of Music Information Retrieval (MIR) to automatically enrich audio archives with metadata. In addition to binary classification where only one of the classes is assumed to be present at time more complex tasks like segmentation of speech or music as well as multi-label classification where multiple classes can be present at time gained popularity. Despite the vast amount of research in this field [23, 12, 14, 24, 13, 5, 20, 4, 8], speech/music classification (SMC) remains challenging in the presence of noise, the involvement of chanting, or under low-quality recording conditions [15]. SMC was first addressed with algorithms based on audio features (e.g., pitch, zero crossing rate) [23, 14, 12]. Recent approaches almost entirely focus on deep neural networks (DNN) that directly learn to detect desired audio properties from input signals and its corresponding annotations [13, 2, 5, 20]. In an attempt to make audio classifiers more robust to varying signal conditions and data scarcity, pre-trained feature representations (embeddings) from related tasks are tranferred to new tasks, so called Transfer Learning (TL), to avoid exhaustive training from scratch [3, 6, 8, 9, 2].

⋆ This work has been supported by the German Research Foundation (BR 1333/20-1, CA 2096/1-1)

This work is divided in two stages. First, we analyze three state-of-the-art neural network architectures for SMC and evaluate their robustness to varying signal conditions by using a diversity of datasets. Here we aim to understand whether any of the three architectures is more robust to varying signal characteristics when trained under comparable conditions. In the second stage of our work, audio embeddings are computed from the three pre-trained architectures. These embeddings are then transferred to different MIR tasks. In this stage, we aim to understand how pre-trained models compare to baseline networks trained from scratch, and whether a close relation of a downstream task and pre-training task exhibit higher learning effects than general audio embeddings like OpenL3 [3] that were not trained on a related MIR task at all.

2 Related Work

Current approaches for SMC mostly rely on deep neural networks (DNN) trained and optimized using raw audio data or its time-frequency transform. The most popular networks for this task are convolutional neural networks (CNN) [12, 13, 5, 20]. In 2015 Lidy et. al [13] used a CNN approach consisting of one convolutional layer followed by a fully connected layer achieving 99.7% accuracy on binary classification of speech and music at the MIREX competition [18]. The separate detection of both classes still achieved 88.5% accuracy. The model proposed by Marolt [15] obtained an accuracy of 98% for SMC, and 92% for a 4-class classification for speech, solo singing, choir, and instrumental music. The model uses a combination of convolutional layers followed by residual layers. Besides the `GTZAN` [25] and `MUSAN` [24] datasets, additional field recordings and traditional music from various libraries were included. In [4], different architectures including DNNs, CNNs and recurrent neural networks were evaluated for speech music detection. According to their findings, a model with six CNN layers performed best on AudioSet [21] with 86% accuracy for speech or music detection. SwishNet [8] uses a set of one-dimensional convolutions with multiple skip connections on Mel-Frequency Cepstral Coefficients (MFCCs). This model achieved 93% accuracy on a 3-class detection task with speech, music, and noise and 99% accuracy for speech detection using the `MUSAN` [24] dataset for training and `GTZAN` [25] for verification. For performance comparison Hussain et al. used a Gaussian Mixture Model, a fully connected neural network (FCN), and a transfer learning approach of the MobileNet architecture [7] was used. The MobileNet embeddings worked best throughout the paper followed by the proposed SwishNet architecture.

Choi et al. [2] showed that transfer learning can outperform traditional feature based methods in many different MIR tasks as well as audio event detection (AED). In [3] OpenL3 embeddings were trained on the task of audio-video correspondence in a self-supervised manner inspired by [1] and subsequently transferred to the task of environmental sound classification. On several AED datasets this approach outperformed other TL embeddings based on VGG-like and SoundNet architectures. Grollmisch et al. [6] verified the potential of OpenL3 for different MIR and industrial sound analysis tasks. The embeddings consistently resulted in good classification performance while other embeddings highly varied depending on the task. Kong et al. [11] proposed pre-trained audio neural networks (PANN) for transfer learning. The authors introduce an input rep-

resentation called Wavegram, a neural network based time-frequency-transformation. A multi-layer CNN is connected to this input network and trained for audio tagging on the AudioSet [21]. Subsequently, these embeddings were augmented by trainable classifiers and applied to six different classification tasks including genre and acoustic scenes classification, among others. In most of these tasks, the embeddings performed better or similar to state-of-the-art approaches. The authors compared multiple networks and depths as well as different positions for unfreezing of the pre-trained embeddings concluding that a complete fine-tuning of all network parameters results in the highest accuracy. To overcome the overfitting to one particular task Kim et al. [9] proposed multi-task learning. During training, a CNN network structure is split at one stage in the model into multiple branches, one for each task. All branches consist of the same network architecture and where trained simultaneously. The last layers before the classifiers of each branch are concatenated and used as combined embeddings. Initially the system was trained on the Million Song Database [16] for tempo estimation and song similarity. The embeddings were evaluated on target tasks like genre classification or music recommendation. Different branch positions in the network were evaluated concluding that earlier branching results in better performance for the target tasks but also in bigger networks with more computational costs.

3 Datasets

To get a better understanding of the performance of the evaluated architectures, four datasets were used during training as depicted in table 1. The `MUSAN` dataset [24] and the `GTZAN` dataset [25] consist of clearly distinguishable broadcast material of western music and speech. In addition, two more challenging ethnomusicology datasets are included. The `Marolt19` dataset was first introduced in [15]. Apart from the speech class, choir, solo singing and instrumental music are combined into the 'music' class for training. `Marolt19` includes material from archives such as the British Library world & traditional music collection, the French Centre of Scientific Research (CNRS), or the Slovenian sound archive Ethnomuse. The `ACMus Youtube Dataset (ACMusYT)`[4] was collected as part of the ACMus research project. [5] It consists of audio excerpts of

[4] `https://zenodo.org/record/4870820`

[5] ACMus project page: `https://acmus-mir.github.io/`

Table 1. Characteristics of the datasets used for training on speech/music classification (source task) and for transfer learning tasks (target tasks).

Application	Dataset ID	Classes [Number of Files per class]	Sample Rate	Bit Depth	Duration [min]
Training	MUSAN	Music [660], Speech [426], Noise [764]	16 kHz	16	6483
	GTZAN	Music [64], Speech [64]	22 kHz	16	64
	Marolt19	Solo Singing [1512], Choir [1618], Instrumental [2960], Speech [1284]	44 kHz	16	577
	ACMusYT	Speech [40], Music [35], A Cappella [40]	48 kHz	16	88
Transfer	S&S	Music [101], Speech [80]	22 kHz	16	45
	ACMusVF	Male [46], Female [24]	96 kHz	24	26
	ACMusIF	1 [43], 2 [42], 3 [43], 4 [21], 5+ [36]	96 kHz	24	65

traditional Colombian music from the Andes region. The subset used in this work consists of two classes: speech and music with vocals. The 'vocal-only' class is not used in these experiments for better separation during training. For TL experiments, the pretrained networks are subsequently fine-tuned with separate datasets. An established set for speech music tasks is the Slaney & Scheirer dataset (S&S) [23] with content taken from broadcast material. All 64 files of noise and mixed (speech/music) content are excluded before the evaluation. From the `ACMus-MIR` dataset [17], the `Instrumental Format Set (ACMusIF)` was used. This set was created from traditional Andean music recordings for the purpose of ensemble size classification. The goal of this task is to classify music tracks as solo, duo, trio, quartet, and larger ensembles. Finally, the `ACMus Vocal Format Set (ACMusVF)` is included.[6] It comprises Andean vocal music (male and female singers) partly with accompaniment.

4 Methodology

4.1 Network Architectures

The INA (Institut National de l'Audiovisuel) approach [5] is a CNN-based network that uses 68 frames of 21 MFCCs with a maximum frequency of 4 kHz as input representation to four 2D-convolutional layers followed by four dense layers with dropout. Each of these layers are followed by batch normalization and a $ReLU$ activation. The output layer uses $Softmax$ activation (see Figure 1 for details). INA achieved an average accuracy of 92.6% at the 2018 MIREX [19] competition on music detection and 96.2% on speech detection.

SwishNet is an architecture based on one-dimensional convolutional layers in combination with residual and skip connections [8] (see Figure 2). As input, 16 frames of 22 MFCCs are extracted from one second audio snippets and used as 2D feature representation. Classification results range from 93% frame-wise accuracy for 3 classes (speech, music, noise) to 99% segment-wise accuracy for speech detection.

VGG-like architectures are commonly used networks in many fields of deep learning [15, 3, 2]. The network illustrated in Figure 3 is inspired by [22]. Logarithmic Mel-Spectrogram (MelSpec) is used as input from audio sampled at 22050 Hz. Frames of 2048 samples with 512 samples hop size are transformed to 128 mel band representation. A patch of 10 frames is fed to four convolutional layers with 32 kernels of size 3x3.

[6] `https://zenodo.org/record/4791394`

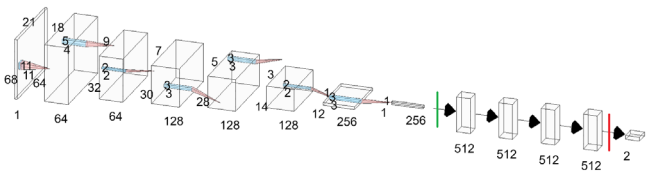

Fig. 1. INA network architecture [5]. The green line indicates the freezing point of the intermediate fine-tuning strategy. The red line indicates the output point of the embedding vector.

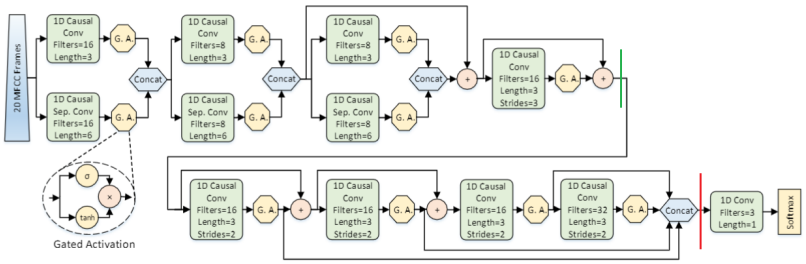

Fig. 2. SwishNet network architecture. The green line indicates the freezing point of the intermediate fine-tuning strategy. The red line indicates the output point of the embedding vector. Refer to [8] for more details on the architecture.

Each layer is followed by batch normalization and ReLU activation. After every second convolutional layer, MaxPooling is applied with a 3x3 window. Two fully connected layers are added after flattening followed by the classifier with a $Softmax$ activation.

OpenL3 embeddings are included as a state-of-the-art baseline. The 512 unit feature vectors are extracted from the audio data with default parameters from [3]. These vectors are normalized between 0 and 1 and used as input for a trainable neural classifier consisting of a 128 unit dense layer followed by the final classifier with $Sigmoid$ activation. As a second baseline, a simple DNN architecture is used. MelSpecs with equal measures as for SwishNet and VGG-like models are input and passed through one dense layer with 128 units and the output layer. The same structure is used for the appended classifiers of the computed embeddings in Section 4.4 and hence gives an insight into the learning effects of the preceded architectures. $Adam$ is used as optimization and $Softmax$ as activation function.

4.2 Input Representation

All datasets were normalized in a range of [-1, 1] in time domain and unified to a sampling rate of 22050 Hz and 16 bits. The MelSpec representation with 128 bands and 512 hop size is evaluated as input representation for all networks. Additionally the original MFCC input representations of the SwishNet and INA approach are included to check for side effects of the input adaption. The original VGG-like approach already

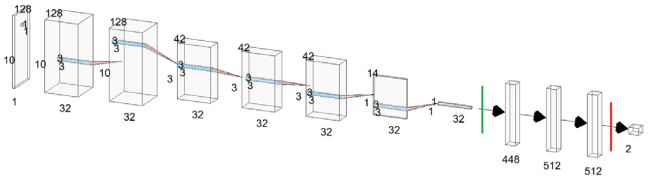

Fig. 3. VGG-like network architecture. The green line indicates the freezing point of the intermediate fine-tuning strategy. The red line indicates the output point of the embedding vector.

used MelSpecs. The OpenL3 embeddings create batches of features with a feature size of 512 samples (see Section 4.1) from 100 ms audio frames.

4.3 Implementation Details and Metrics

In all experiments, 10% of the data is used for testing, and 10% for validation. All experiments are repeated using five-fold cross-validation. All data is balanced by random down-sampling. After transforming the input to MelSpec, it is normalized feature-wise to zero mean in the range from -1 to 1 and concatenated to batches of 64 frames. Each network is trained for 200 epochs with the option for early stopping if the validation accuracy does not increase for 50 epochs. The $Adam$ optimizer [10] with a learning rate of 10^{-3} is used for all architectures for best comparability to the original implementations. Results are presented as the mean accuracy over 5 cross-validation folds with its standard deviation.

4.4 Transfer Learning Networks and Tasks

For transfer learning, the models are trained with a balanced combination of all four training sets. Afterwards the output layers are removed from the trained networks (see Section 4.1) and the remaining layers are fixed and used for embedding calculation. A trainable classifier is appended consisting of a 128 unit dense layer and a dense output layer matching the number of the target task classes. Three different freezing positions for the trained models are evaluated. In the first strategy, only the classifier is trained while the network weights remain fixed. The second strategy unfreezes the networks in an intermediate position so the classifier and parts of the networks are fine-tuned. These positions are illustrated green in Figures 1, 2, and 3, respectively. In a third strategy, all network weights are unfrozen and fine-tuned along with the classifier. These strategies do not apply for OpenL3 because of its baseline function. As transfer learning tasks, we evaluate the following target tasks: (a) SMC with `S&S` dataset, (b) accompaniment detection with `ACMusVF` dataset. The goal of this task is to distinguish music pieces with instrumental accompaniment from vocal-only performances, (c) female vs male singer classification on the `ACMusVF` dataset. We refer to this task as gender classification in singing, (d) ensemble size classification on the `ACMusIF` set.

5 Results

5.1 Network Architectures Comparison

Figure 4 shows the mean file-wise and frame-wise results over all training sets for each architecture. Results show that OpenL3 embeddings work well on all datasets for SMC. Looking at the frame-wise accuracy, SwishNet is slightly below the remaining two CNN-based architectures by around 3%. Figure 5 presents results for binary SMC and a three-class task which includes noise as the third class. This is performed for the `MUSAN` and `Marolt19` datasets where noise samples are included. `Marolt19` appears to be the most challenging set due to the fact that it does not only consist of

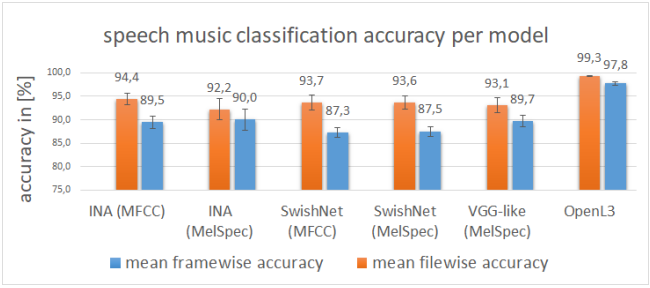

Fig. 4. Comparison of the mean frame-wise accuracy per architecture for speech/music classification averaged over all training sets (`MUSAN, GTZAN, Marlot19, ACMusYT`).

broadcast material unlike `MUSAN`. As expected, the accuracy drops for a more complex task of three classes. The highest drop of 24.3 % occurs for INA in connection with MelSpec input followed by the VGG-like model. For `MUSAN` the most significant drop can be observed for the INA model in connection with MFCC input. The varying results indicate that the INA architecture might not be well suited for alternative tasks in contrast to OpenL3 which shows best robustness. Regarding the input representation no significant performance differences can be observed in Figure 4. Only a slight improvement for MelSpecs is visible. Figure 5 confirms this trend as MelSpecs have a slightly better performance on average. In conclusion MFCCs can increase performance for specific tasks but MelSpecs have a more robust behavior in general hence MelSpec is used for further experiments.

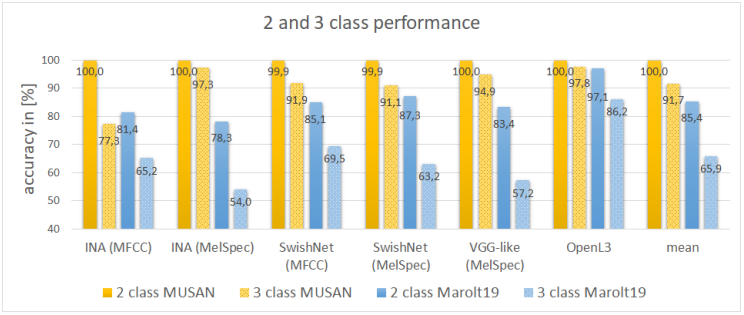

Fig. 5. Comparison of frame-based accuracy for binary classification versus 3-class classification. Results are shown for `MUSAN` (yellow) and `Marolt19` (blue) datasets.

5.2 Transfer Learning

Results for all transfer experiments are presented in Table 2. Besides the three network architectures (INA, SwishNet, and VGG-like), results for the OpenL3 embeddings and

the DNN baselines are shown. In general the resulting models tend to overfit during fine-tuning due to the small training data.

Speech Music Classification with S&S: In this experiment, the target task for TL was kept the same so models are only transferred to an unseen dataset. In Table 2 a learning effect from the pre-training can be observed for the Slaney & Scheirer dataset. In detail embeddings from INA and VGG-like models can make better use of pre-training and gain up to 3 % classification accuracy while the performance of SwishNet remains at almost the same level. OpenL3 embeddings outperform all other models for this dataset-task combination.

Table 2. Transfer learning results. Accuracy values are presented for fully frozen (Acc_{FZ}), partly trainable (intermediate) (Acc_{IN}), and the fully trainable embeddings (Acc_{FT}). Listed are the results for each architecure using their pre-trained embeddings (*Emb*) as well as their original network trained from scratch on the according task (*Orig*). In addition OpenL3 embeddings and the two-layer DNN (see 4.1) are listed as baseline.

Task-Set-Combination	Model	Acc_{FZ} [%]	Acc_{IN} [%]	Acc_{FT} [%]
Speech Music on S&S	INA_{Emb}	98,8 ± 1,4	97,6 ± 2,1	85,1 ± 4,8
	INA_{Orig}	-	-	93,8 ± 3,0
	$VGG-like_{Emb}$	97,4 ± 1,5	97,9 ± 1,9	88,9 ± 1,6
	$VGG-like_{Orig}$	-	-	95,1 ± 2,1
	$SwishNet_{Emb}$	92,3 ± 2,7	93,0 ± 2,4	95,0 ± 1,7
	$SwishNet_{Orig}$	-	-	92,9 ± 1,5
	$OpenL3_{Emb}$	**99,2** ± 0,4	-	-
	$DNN_{baseline}$	-	-	92,9 ± 1,9
Accompaniment on ACMus VF	INA_{Emb}	85,2 ± 5,6	82,5 ± 9,1	90,8 ± 5,2
	INA_{Orig}	-	-	80,2 ± 6,6
	$VGG-like_{Emb}$	88,5 ± 7,4	94,9 ± 3,2	92,7 ± 5,8
	$VGG-like_{Orig}$	-	-	92,7 ± 4,9
	$SwishNet_{Emb}$	81,5 ± 4,6	85,1 ± 4,6	93,6 ± 3,2
	$SwishNet_{Orig}$	-	-	94,0 ± 3,7
	$OpenL3_{Emb}$	**99,6** ± 0,5	-	-
	$DNN_{baseline}$	-	-	96,5 ± 1,7
Gender on ACMus VF	INA_{Emb}	70,0 ± 7,7	47,3 ± 7,9	59,3 ± 7,6
	INA_{Orig}	-	-	67,4 ± 7,0
	$VGG-like_{Emb}$	71,8 ± 5,2	75,8 ± 9,1	73,5 ± 6,2
	$VGG-like_{Orig}$	-	-	73,6 ± 8,1
	$SwishNet_{Emb}$	72,6 ± 5,0	73,1 ± 5,1	**78,3** ± 8,9
	$SwishNet_{Orig}$	-	-	74,9 ± 9,5
	$OpenL3_{Emb}$	72,3 ± 9,6	-	-
	$DNN_{baseline}$	-	-	72,6 ± 10,3
Ensemble Size on ACMus IF	INA_{Emb}	49,8 ± 5,6	52,1 ± 10,2	56,7 ± 4,5
	INA_{Orig}	-	-	48,8 ± 7,2
	$VGG-like_{Emb}$	49,7 ± 5,0	51,3 ± 6,8	47,1 ± 3,9
	$VGG-like_{Orig}$	-	-	57,9 ± 5,3
	$SwishNet_{Emb}$	46,7 ± 5,7	48,7 ± 6,3	54,3 ± 5,4
	$SwishNet_{Orig}$	-	-	56,3 ± 5,6
	$OpenL3_{Emb}$	**76,2** ± 4,4	-	-
	$DNN_{baseline}$	-	-	61,4 ± 5,3

Accompaniment detection on ACMusVF: For this task OpenL3 again shows best results and is followed by the VGG-like embeddings with a performance gap of around 11 %. Despite the close task relation to SMC no architecture overcomes the accuracy of the plain DNN and hence no learning effect from TL is achieved in connection with

this task. This is reinforced by the fact that for SwishNet and VGG-like architectures, the original models perform better than their embedding counterparts.

Female/Male singer classification on ACMusVF: For this task SwishNet embeddings show best results closely followed by OpenL3 embeddings. The original networks for each model show comparable or better performances compared to the fully frozen embeddings indicating that no learning effect of pre-training is visible. Again the DNN performs comparable to the best model refuting a benefit of the knowledge transfer.

Ensemble size classification on ACMus-MIR: All created embeddings perform similar with nearly 50 % accuracy. The baseline architectures of VGG-like and SwishNet show better results when trained from scratch excluding the idea of a possible learning effect. This is confirmed by the plain DNN baseline that outperformed the embeddings by around 12 %. The usage of embeddings results in a inverse effect for this task. Furthermore this experiment engages the most unrelated task relative to SMC in the set of transfer tasks. The best results are achieved using the unrelated OpenL3 embeddings with 76.2 %. A file-wise evaluation of OpenL3 results in 84 % accuracy which confirms the outcome from Grollmisch et al. [6].

Freezing strategies: Inspecting the last two rows of each embedding in table 2 gives insights to freezing strategies for the pre-trained networks. With more degree of freedom, meaning more trainable layers, the accuracy tend to increase in most cases. This trend is highly network-dependent and mainly applies to SwishNet models while INA tends to be more unstable showing a higher fluctuation. VGG-like models perform best in intermediate state.

6 Conclusions

This work examines the idea of transfer learning (TL) by creating new feature representations from one source task (pre-training), to use them as embeddings for several target MIR tasks. Three network architectures (INA, SwishNet, VGG-like) were initially trained for SMC, and subsequently applied to four new classification tasks. Our experiments show a slight dominance of the MelSpec as input representation over MFCCs during training. No significant performance difference between the three architectures is visible for the source task while OpenL3 embeddings consistently showed best SMC accuracy. In comparison to the networks trained from scratch, pre-training results in a slight improvement when used with an additional DNN classifier for the source task. In the TL experiments, the direct combination of MelSpec input and the DNN classifier surpasses the embedding performance in some cases. These results suggest that the learning effect of pre-training is not consistent over all experiments. Furthermore, creating embeddings with tasks closely related to the target tasks show no evident benefit compared to general audio embeddings such as OpenL3, which performed best in most of the cases. A possible cause can be the self-supervised creation of these embeddings which inhabits limitless availability of training data. However, the amount of training data used for pre-training the different embeddings is not considered in these experiments and is left for future work.

References

1. R. Arandjelovic and A. Zisserman. Look, listen and learn. *CoRR*, 2017.
2. K. Choi, G. Fazekas, M. Sandler, and K. Cho. Transfer learning for music classification and regression tasks. *CoRR*, 2017.
3. J. Cramer, H. Wu, J. Salamon, and J. Bello. Look, listen, and learn more: Design choices for deep audio embeddings. In *ICASSP*, 2019.
4. D. de Benito, A. Lozano-Diez, D. Toledano, and J. Gonzalez-Rodriguez. Exploring convolutional, recurrent, and hybrid deep neural networks for speech and music detection in a large audio dataset. *EURASIP*, 2019.
5. D. Doukhan, E. Lechapt, M. Evrard, and J. Carrive. Ina's mirex 2018 music and speech detection system. In *MIREX 2018*, 09 2018.
6. S. Grollmisch, E. Cano, C. Kehling, and M. Taenzer. Analyzing the potential of pre-trained embeddings for audio classification tasks. In *28th EUSIPCO*, 2021.
7. A. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, and M. Andreetto. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017.
8. S. Hussain and M. Ariful Haque. Swishnet: A fast convolutional neural network for speech, music and noise classification and segmentation, 2018.
9. J. Kim, J. Urbano, C. Liem, and A. Hanjalic. One deep music representation to rule them all? : A comparative analysis of different representation learning strategies. *CoRR*, 2018.
10. Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
11. Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. Plumbley. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *CoRR*, 2019.
12. A. Kruspe, D. Zapf, and H. Lukashevich. Automatic speech/music discrimination for broadcast signals. *LNI Proceedings*, 2017.
13. T. Lidy. Spectral convolutional neural network for music classification. *MIREX*, 2015.
14. M. Marolt. Probabilistic segmentation and labeling of ethnomusicological field recordings. *Proceedings of ISMIR*, 2009.
15. M. Marolt, C. Bohak, A. Kavcic, and M. Pesek. Automatic segmentation of ethnomusicological field recordings. *Applied Sciences*, 2019.
16. millionsongdataset.com. Welcome! — million song dataset.
17. F. Mora-Ángel, G. López Gil, E. Cano, and S. Grollmisch. ACMUS-MIR: An Annotated Dataset of Andean Colombian Music. In *7th DLFM Conference*, 2019.
18. music ir.org. 2015:music/speech classification and detection results - mirex wiki.
19. music ir.org. 2018:music and or speech detection results - mirex wiki.
20. M. Papakostas and T. Giannakopoulos. Speech-music discrimination using deep visual feature extractors. *Expert Systems with Applications*, 2018.
21. research.google.com. Audioset.
22. Y. Sakashita and M. Aono. Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions. Technical report, DCASE2018 Challenge, 2018.
23. E. Scheirer and M. Slaney. Construction and evaluation of a robust multifeature speech/music discriminator. *ICASSP'97*, 1997.
24. D. Snyder, G. Chen, and D. Povey. Musan: A music, speech, and noise corpus, 2015.
25. G. Tzanetakis. marsyas.info gtzan speech music dataset download. `http://opihi.cs.uvic.ca/sound/music_speech.tar.gz`.