Contrastive Learning of Musical Representations

Spijkervet, J.; Burgoyne, J.A.

DOI
10.5281/zenodo.5624573

Publication date
2021

Document Version
Final published version

Published in
Proceedings: The 22nd International Society for Music Information Retrieval Conference

CONTRASTIVE LEARNING OF MUSICAL REPRESENTATIONS

Janne Spijkervet **John Ashley Burgoyne**
Institute for Logic, Language, and Computation, University of Amsterdam
janne.spijkervet@gmail.com, j.a.burgoyne@uva.nl

ABSTRACT

While deep learning has enabled great advances in many areas of music, labeled music datasets remain especially hard, expensive, and time-consuming to create. In this work, we introduce SimCLR to the music domain and contribute a large chain of audio data augmentations to form a simple framework for self-supervised, contrastive learning of musical representations: CLMR. This approach works on raw time-domain music data and requires no labels to learn useful representations. We evaluate CLMR in the downstream task of music classification on the MagnaTagATune and Million Song datasets and present an ablation study to test which of our music-related innovations over SimCLR are most effective. A linear classifier trained on the proposed representations achieves a higher average precision than supervised models on the MagnaTagATune dataset, and performs comparably on the Million Song dataset. Moreover, we show that CLMR's representations are transferable using out-of-domain datasets, indicating that our method has strong generalisability in music classification. Lastly, we show that the proposed method allows data-efficient learning on smaller labeled datasets: we achieve an average precision of 33.1% despite using only 259 labeled songs in the MagnaTagATune dataset (1% of the full dataset) during linear evaluation. To foster reproducibility and future research on self-supervised learning in music, we publicly release the pre-trained models and the source code of all experiments of this paper.

1. INTRODUCTION

Supervised learning methods have been widely used in musical tasks like chord recognition [1, 2], key detection [3], beat tracking [4], music audio tagging [5] and music recommendation [6]. These methods require labeled corpora, which are difficult, expensive and time-consuming to create for music in particular [7], while raw unlabeled music data is available in vast quantities. Unsupervised alternatives to end-to-end deep learning for music are compelling, especially if they can generalise to smaller datasets.

Despite the importance of unsupervised learning for raw audio signals, unsupervised learning for musical tasks has yet to see breakthroughs comparable to those in supervised learning. There have been successes with methods like PCA, PMSC's and spherical k-means that rely on a transformation pipeline [8, 9], and very recently with self-supervised methods in the time-frequency domain for general audio classifiation tasks [10–13], but learning effective representations of raw audio in an unsupervised manner has remained elusive for musical tasks.

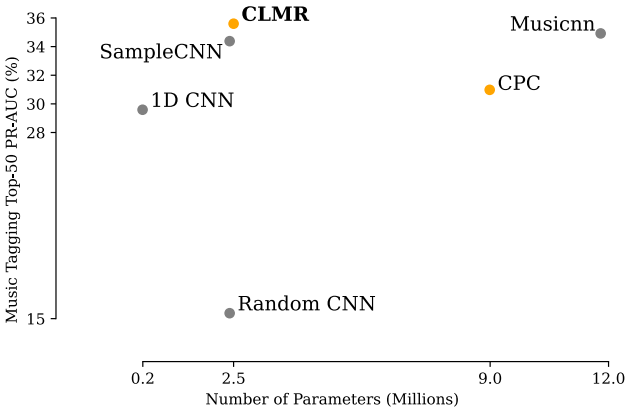

Figure 1: Performance and model complexity comparison of supervised models (grey) and self-supervised models (ours) in music classification of raw audio waveforms on the MagnaTagATune dataset to evaluate musical representations. Supervised models were trained end-to-end, while CLMR and CPC are pre-trained without ground truth: their scores are obtained by training a *linear* classifier on their learned representations but nonetheless perform competitively to the supervised models.

Self-supervised representation learning is an unsupervised learning paradigm that has demonstrated advances across many tasks and research domains [14–18]. This includes the ability to use substantially less labeled data when fine-tuning on a specific task [17, 19, 20]. Without ground truth, there can be no ordinary loss function for training; self-supervised learning trains by way of a proxy loss function instead. One way to preserve the amount of useful information during self-supervised learning is to define the proxy loss function with respect to a relatively simple pretext task, with the idea that a representation that is good for the pretext task will also be useful for downstream tasks. Many approaches rely on heuristics to design pretext tasks [21, 22], e.g., by witholding a pitch transformation [23]. Alternatively, *contrastive representation learning* formulates the proxy loss directly on the learned

representations and relies on contrasting multiple, slightly differing versions of any one example by often using negative sampling strategies [17,24,25] or by bootstrapping the representations [18].

In this paper, we combine the insights of a simple contrastive learning framework for images, SimCLR [17], with recent advances in representation learning for audio in the time domain [26]. We also contribute a pipeline of data augmentations on musical audio, to form a simple framework for self-supervised, contrastive learning of representations of raw waveforms of music. To compare the effectiveness of this simple framework compared to a more complex self-supervised learning objective, we also evaluate representations learned by contrastive predictive coding (CPC) [15]. The self-supervised models are evaluated on the downstream music tagging task, enabling us to evaluate their versatility: music tags describe many characteristics of music, e.g., genre, instrumentation and dynamics. Our key contributions are the following.

- CLMR achieves strong performance on the music classification task compared to supervised models, despite self-supervised pre-training and training a linear classifier on the downstream task with raw signals of musical audio (see Figure 1).
- CLMR enables efficient classification: we achieve comparable performance using as few as 1% of the labeled data.
- We show the out-of-domain transferability of representations learned from pre-training CLMR on entirely different corpora of musical audio.
- CLMR can learn from *any* dataset of raw music audio, requiring neither transformations nor fine-tuning on the input data; nor do the models require manually annotated labels for pre-training.
- We provide an ablation study on the effectiveness of individual audio data augmentations.

2. RELATED WORK

The goal of representation learning is to identify features that make prediction tasks easier and more robust to the complex variations of natural data [27]. In unsupervised representation learning, generative modeling and likelihood-based models typically find useful representations of the data by attempting to reconstruct the observations on the basis of their learned representations [28, 29]. *Self-supervised* representation learning aims to identify the explanatory factors of the data using an objective that is formulated with respect to the learned representations directly [15, 18, 19, 21, 22].

Compared to vision, work on self-supervised learning in audio is still very limited, but there are a number of works that appeared very recently. Contrastive predictive coding is a universal approach to contrastive learning, and has been successful for speaker and phoneme classification using raw audio, among other tasks [15]. PASE [30] introduces several self-supervised workers that solve regression or binary discrimation tasks, that jointly optimise an encoder for speech recognition. To improve the representations for mismatched acoustic conditions and their transferability, they apply augmentations to the input speech signal [31]. In music information retrieval, recent advances have been made in self-supervised pitch estimation [23], closely matching supervised, state-of-the-art baselines [32] despite being trained without ground truth labels. L^3-Net learns deep embeddings from audio-visual correspondence in videos by way of self-supervised learning [10]. Their work uses mel-spectrograms for audio and requires more than 40 million audio-video training samples to learn optimal embeddings. Audio2Vec also operates in the time-frequency domain and learns by reconstructing spectrogram slices from past and future slices [11]. With limited data, Audio2Vec outperforms supervised models in pitch and instrument classification. CLAR also uses a contrastive learning objective, and computes a loss on a concatenation of representations learned from both raw audio and mel-spectrograms [12]. COLA uses a similar method with mel-spectrograms only, and uses bilinear comparisons instead of cosine similarity [13]. Both works are evaluated on speech command, environmental sound classification, and on pitch and instrument classification on the NSynth dataset [33].

3. METHOD

This work builds on SimCLR, a simple contrastive learning framework of visual representations [17]. Despite a task-agnostic, labelless discriminative pre-training approach, a linear classifier achieved performance comparable to fully supervised models in many image classification benchmarks. Its learning objective is to maximise the agreement of latent representations of augmented views of the same image using a contrastive loss. In Section 2, we will continue an overview of contrastive learning.

In CLMR, we adapt this framework to the domain of raw music audio. While most core components of CLMR have appeared in previous work, its ability to model waveforms of music cannot be explained by a single design choice, but by their composition. We will first elaborate the four core components in the following subsections:

- A stochastic composition of data augmentations that produces two correlated, augmented examples of the same audio fragment, the 'positive pair', denoted as x_i and x_j.
- An encoder neural network $g_{\text{enc}}(\cdot)$ that maps the augmented examples to their latent representations.
- A projector neural network $g_{\text{proj}}(\cdot)$ that maps the encoded representations to the latent space where the contrastive loss is formulated.
- A contrastive loss function, which aims to identify x_j from the negative examples in the batch $\{x_{k\neq i}\}$ for a given x_i.

The complete framework is visualised in Figure 2.

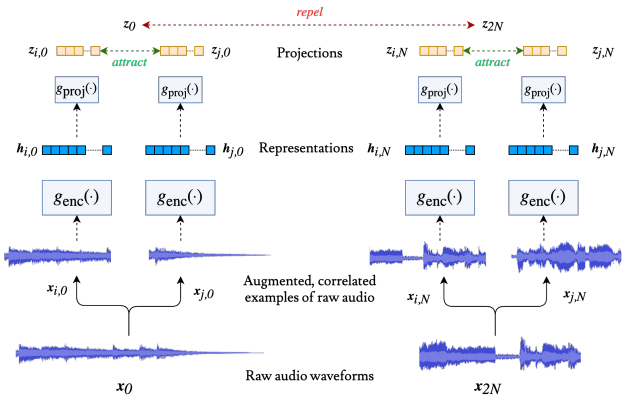

Figure 2: The complete framework operating on raw audio, in which the contrastive learning objective is directly formulated in the latent space of correlated, augmented examples of pairs of raw audio waveforms of music.

3.1 Data Augmentations

We designed a comprehensive chain of audio augmentations for raw audio waveforms of music to make it harder for the model to identify the correct pair of examples. For details, see Appendix B [1] . Each consecutive augmentation is stochastically applied on x_i and x_j independently, i.e., each augmentation has an independent probability $p_{\text{transform}}$ of being applied to the audio. The order of augmentations applied to audio is carefully considered, e.g., applying a delay effect *after* reverberation empirically gives an entirely different result in music.

1. A random fragment of size s is selected from a piece of music, without trimming silence (e.g., the intro or outro of a song). The two examples x_i and x_j from the same audio fragment can overlap or be very disjoint, allowing the model to infer both local and global structures.
2. The polarity of the audio signal is inverted, i.e., the amplitude is multiplied by -1.
3. Additive white Gaussian noise is added with a signal-to-noise ratio of 80 decibels to the original signal.
4. The gain is reduced between $[-6, 0]$ decibels.
5. A frequency filter is applied to the signal. A coin flip determines whether it is a low-pass or a high-pass filter. The cut-off frequencies are drawn from uniform distributions on $[2200, 4000]$ or $[200, 1200]$ Hz respectively.
6. The signal is delayed and added to the original signal with a volume factor of 0.5. The delay is randomly sampled between 200-500ms, in 50ms increments.
7. The signal is pitch shifted. The pitch transposition interval is drawn from a uniform distribution of semitones between $[-5, 5]$, i.e., a perfect fourth compared to the original signal's scale.
8. Reverb is added to alter the signal's acoustics. The impulse response's room size, reverbation and damping factor is drawn from a uniform distribution on $[0, 100]$.

[1] The supplementary material can be found at the accompanying webpage of this paper : `https://spijkervet.github.io/CLMR`

The space of augmentations is not limited to these operations and could be extended to, e.g., randomly applying chorus, distortion and other modulations. Some of these have been shown to improve performance in self-supervised learning for automatic speech recognition in the time-domain as well [31, 34].

3.2 Batch Composition

A larger batch size N makes the contrastive learning objective harder – there are simply more negative examples the anchor sample needs to identify the positive sample from – but it can substantially improve model performance [17]. We sample one song from the batch, augment it into two examples, and treat them as the positive pair. We treated the remaining $2(N-1)$ examples in the batch as negative examples, and did not sample the negative examples explicitly. Larger batch sizes introduces a practical problem for raw audio when training on a GPU, as their input dimensionality increases for higher sample rates. When training on multiple GPU's, we used global batch normalisation, i.e., we aggregate the batch statistics over all devices during parallel training, to avoid potential leakage of batch statistics because the positive examples are sampled on the same device (which improves training loss, but counteracts learning of useful representations).

3.3 Encoder

To directly compare a state-of-the-art end-to-end supervised model used in music classification on raw waveforms against a self-supervised model, we use the SampleCNN architecture as our encoder [26]. Similarly, we use a fixed audio input of 59 049 samples with a sample rate of 22 050 Hz. In this configuration, the SampleCNN encoder g_{enc} consists of 9 one-dimensional convolution blocks, each with a filter size of 3, batch normalisation, ReLU activation and max pooling with pool size 3. The final output layer is removed, which yields a 512-dimensional feature vector h_i for every audio input. The feature vectors from the encoder can be directly used in the learning objective, but formulating the objective on encodings mapped to a different latent space by a parameterised function helps the effectiveness of the representations [17]. In our experiments, we use a non-linear layer $z_i = W^{(2)}\,\text{ReLU}(W^{(1)}h_i)$ with an output dimensionality of 128 as the projection head g_{proj}. There are 2.5 million trainable parameters in total, which is put in comparison with other state-of-the-art models in Figure 1.

We used 96 examples per batch and the aforedescribed encoder configuration to directly compare our self-supervised performance with the equally expressive fully supervised method [26]. We ran experiments with batch sizes of 96 on 2× NVIDIA 1080Ti, while for larger batches up to 4 × Titan RTX's were used. With 2 1080Ti's, it takes ∼5 days to train 1 000 epochs on our largest dataset.

3.4 Contrastive Loss Function

In keeping with recent findings on several objective functions in contrastive learning [17], the contrastive loss function used in this model is normalised temperature-scaled cross-entropy loss, commonly denoted as *NT-Xent loss*:

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(z_i, z_j\right)/\tau\right)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp\left(\text{sim}\left(z_i, z_k\right)/\tau\right)} \quad (1)$$

The pairwise similarity is measured using cosine similarity and the temperature parameter τ helps the model learn from hard negatives. The indicator function $\mathbb{1}_{[k \neq i]}$ evaluates to 1 iff $k \neq i$. This loss is computed for all pairs, both (z_i, z_j) and (z_j, z_i), for $i \neq j$.

3.5 Contrastive Predictive Coding

We adjusted the original CPC encoder g_{enc} [15] to a deeper architecture for more direct comparison [26]. The encoder g_{enc} consists of 7 layers with 512 filters each, and filter sizes $[10, 6, 4, 4, 4, 2, 2]$ and strides $[5, 3, 2, 2, 2, 2, 2]$. Instead of relying on max-pooling, the filter sizes and strides are adjusted to parameterise and facilitate downsampling. We also increased the number of prediction steps to 20, effectively asking the network to predict 100 ms of audio into the future. The batch size is set to 64 from which 15 negative examples in the contrastive loss are drawn.

3.6 Linear Evaluation

The evaluation of representations learned by self-supervised models is commonly done with linear evaluation [15–17], which measures how linearly separable the relevant classes are under the learned representations. We obtain the representations for all datapoints from a frozen CLMR network after pre-training has converged, and train a linear classifier using these self-supervised representations on the downstream task of music classification. For CPC, the representations are extracted from the autoregressor, yielding a context vector of size $(20, 256)$, which is global-average pooled to obtain a single vector of 512 dimensions. For CLMR, the last 512-dimensional vector h from the encoder is used instead of z from the projection head because that yielded consistently better results for all our experiments. We compute the evaluation metrics on a held-out test set, averaged over three runs on the training set using different random seeds.

3.7 Optimisers

We use the Adam optimiser [35] with a learning rate of 0.0003 and $\beta_1 = 0.9$ and $\beta_2 = 0.999$ during pre-training and employ He initialisation for all convolutional layers. The temperature parameter τ is set to 0.5, since we observed consistent results regardless of varying batch sizes and temperature $\tau \in \{0.1, 0.5, 1.0\}$. For linear evaluation, we use the Adam optimiser with a learning rate of 0.0003 and a weight decay of 10^{-6}. Backpropagation is only done in the final (linear) head for all experiments in this paper. We also employ an early stopping mechanism when the validation scores do not improve for 5 epochs.

Model	Dataset	ROC-AUC	PR-AUC
CLMR (ours)	MTAT	88.7 (**89.3**)	35.6 (**36.0**)
Musicnn [5]†	MTAT	89.0	34.9
SampleCNN [26]†	MTAT	88.6	34.4
CPC (ours)	MTAT	86.6 (88.0)	31.0 (33.0)
1D CNN [36]†	MTAT	85.6	29.6
Transformer [37]†§	MSD	**89.7**	**34.8**
Musicnn [5]†	MSD	88.0	28.7
SampleCNN [26]†	MSD	87.9	28.5
CLMR (ours)	MSD	85.7	25.0

Table 1: Tag prediction performance on the MagnaTagATune (MTAT) dataset and Million Song Dataset (MSD), compared with fully supervised models(†) trained on raw audio waveforms. We omit most works that operate on (mel-) spectrograms(§) to make a fair comparison with our approach on raw audio. For reference, we add the Transformer model that is the current state-of-the-art in music tagging. For the self-supervised models, the scores are obtained by training a *linear*, logistic regression classifier using the frozen representations from self-supervised pre-training. Scores in brackets show performance when adding a hidden layer to the linear classifier.

4. EXPERIMENTAL RESULTS

4.1 Datasets

We evaluated the quality of our representations with music classification experiments. Predicting the top 50 semantic tags in the MagnaTagATune and Million Song datasets [38, 39] is a popular benchmark for music classification. These semantic tags are annotated by human listeners, and have a varying degree of abstraction and describe many facets of music, including genre, instrumentation and dynamics. It is a multi-label classification task: each track can have multiple tags, of which we use the 50 most frequently occuring to compare our performance against supervised benchmarks.

The MagnaTagATune dataset consists of 25k music clips from 6 622 unique songs, of which we use about 187k fragments of 2.6 seconds for training, and the same train/test split as previous work [5,9,26]. The Million Song Dataset contains a million songs, of which about 240k previews of 30 seconds are available and labeled with Last.FM tag annotations. We only use the train, validation and test split of 201 680 / 11 774 / 28 435 songs as used in previous work [5, 26], not all million songs during self-supervised pre-training. This results in 2.2 million music fragments of 2.6 seconds for training, i.e., almost 1 600 hours of music. The tags for the Million Song Dataset also contain overlapping genre and semantic tags, e.g., 'beautiful', 'happy' and 'sad', which are arguably harder to separate during the linear evaluation phase.

We use average tag-wise area under the receiver operating characteristic curve (ROC-AUC) and average precision (PR-AUC) scores as evaluation metrics. They are measured globally for the whole dataset, i.e., for the tag metric we measure the retrieval performance on the tag dimension (column-wise) and for the clip metric we measure the

performance on the clip dimension (row-wise). PR-AUC is calculated in addition to ROC-AUC, because ROC-AUC scores can be over-optimistic for imbalanced datasets like MagnaTagATune [40].

4.2 Quantitative Evaluation

The most important goal set out in this paper is to evaluate the difference in performance between an otherwise identical, fully supervised network when learning representations using a self-supervised objective.

CLMR exceeds the supervised benchmark for the MagnaTagATune dataset with a PR-AUC of 35.6%, despite task-agnostic, self-supervised pre-training and a *linear* classifier for training, as shown in Table 1. An additional 0.4% PR-AUC performance gain is added by adding an extra hidden layer to the classifier. When increasing the batch size and the number of parameters, we observe another performance gain to 37.0% PR-AUC as show in Appendix C.1. The performance on the larger Million Song Dataset is lower compared to the supervised benchmark, and especially to the current state-of-the-art model that is trained using mel-spectrograms [37], but is still remarkable given the use of a linear classifier. The tags in the Million Song Dataset are semantically more complex, e.g., 'catchy', 'sexy', 'happy', and have more similar genre tags, e.g., 'progressive rock', 'classic rock' and 'indie rock', which our proposed contrastive learning method may not distinguish.

CPC also shows competitive performance with fully supervised models in the music classification task. Despite CPC's good performance, self-supervised training does not require a memory bank or more complex loss functions, e.g., those incorporating mutual information or more explicit negative sampling strategies, to learn useful representations.

We also analyse the quality of our representations, showing they can cleanly separate audio fragments from different classes, and visualise the convolution filters of the self-supervised models in Appendix C.4.

4.3 Data Augmentations

The CLMR model relies on a pipeline of strong data augmentations to facilitate the learning of representations that are more robust and allow for better generalisation in the downstream task. In Table 2, we show the linear evaluation scores obtained by taking a random crop of audio and performing one additional, individual augmentation. While all datasets contain songs of variable length, we always sample a random crop of audio of the same size before applying other augmentations. This makes it harder to assess the individual contribution of each augmentation to the downstream task performance. We therefore consider an asymmetric data transformation setting: we only apply the augmentation(s) to one branch of the framework, while we settle with an identity function for the other branch (i.e., $t(x_j) = x_j$) [17]. The model is pre-trained from scratch for 1 000 epochs after which linear evaluation is performed.

	Tag		Clip	
Transform	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC
Filter	87.6	33.3	92.5	67.9
Reverb	86.5	31.7	91.8	65.8
Polarity	86.3	31.5	91.7	65.7
Noise	86.1	31.5	91.5	65.5
Pitch	86.4	31.5	91.5	65.3
Gain	86.2	31.1	91.5	65.1
Delay	85.8	30.5	91.3	64.9
Crop	85.8	30.5	91.3	64.8

Table 2: CLMR music tagging performance using a random crop together with one other audio data augmentation.

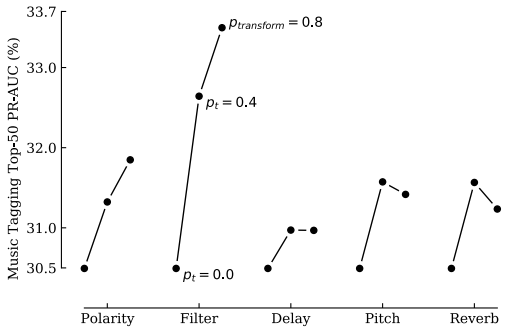

Figure 3: $\mathrm{PR-AUC_{TAG}}$ scores for transformations under different, consecutive probabilities $p \in \{0.0, 0.4, 0.8\}$

When only taking a random crop of audio, we achieve a PR-AUC score of 30.5. Most individual augmentations show an increase in performance, while adding gain or delay does not impact performance as much. Adding a filter to the augmentation pipeline increases the downstream performance more significantly.

Besides evaluating the individual contribution of each augmentation with augmentation probability $p_t = 1$, we also vary $p_t \in \{0, 0.4, 0.8\}$. This is done to assess the optimal amount of augmentation to each example, i.e., the contrastive learning task should neither be too hard, nor too simple, for learning effective representations for the music classification task. The linear evaluation PR-AUC score is shown for each augmentation under a different probability p_t in Figure 3. For the Polarity and Filter transformations, performing them more often with a probability of $p_t = 0.8$ is beneficial. For the Delay, Pitch and Reverb transformations, a transformation probability of $p_t = 0.4$ works better than performing them more aggressively. Generally, we find that strong data augmentations result in more robust representations and better downstream task performance.

4.4 Data Efficient Classification Experiments

To test the efficient classification capability of the CLMR model, we train the linear classifier on consecutive, class-balanced subsets of the labels in the train dataset and report its performance. During the task-agnostic, self-supervised pre-training phase, 100% of the data is used. Figures 4 and 5 show the PR-AUC scores obtained when increasing

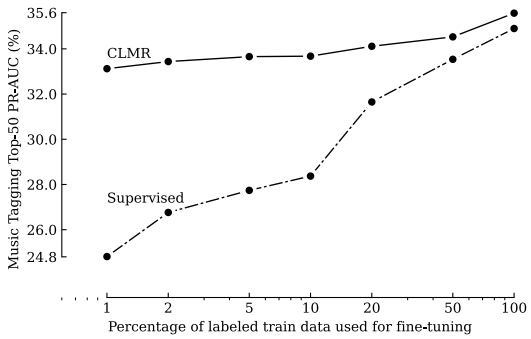

Figure 4: Percentage of labels used for training vs. the achieved $\mathrm{PR}-\mathrm{AUC}_{\mathrm{TAG}}$ score on the MTAT dataset.

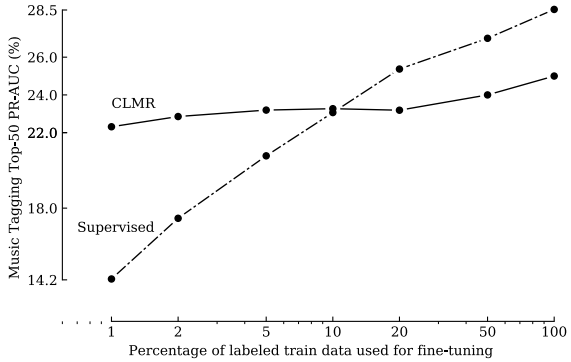

Figure 5: Percentage of labels used for training vs. the achieved $\mathrm{PR}-\mathrm{AUC}_{\mathrm{TAG}}$ score on the MSD.

the amount of labels available during training. For both datasets, self-supervised pre-training greatly improves performance when less labeled data is available. Using 100 times fewer labeled songs, i.e., only 259 songs, CLMR scores 33.1% PR-AUC compared to 24.8% PR-AUC obtained with an equivalent, end-to-end trained supervised model trained on about 25 000 songs. Pre-training using a self-supervised objective without labels therefore substantially improves efficient classification: only 1% of the labels are required while maintaining a similar performance. For the Million Song Dataset, a fully supervised model exceeds CLMR at 10% of the labels, which are 24 190 unique songs in total.

4.5 Transfer Learning Experiments

To test the out-of-domain generalisability of the learned representations, we pre-trained CLMR on entirely different music datasets. After pre-training, we freeze the weights of the network, i.e., we do not fine-tune the encoder, and subsequently perform the linear evaluation procedure outlined in Section 3.6. While originally made for chord recognition, we use 461 contemporary pop songs recorded between the 1940's and 2000's from the McGill Billboard dataset [41]. The Free Music Archive dataset [42] consists of 22 413 songs for the 'medium' version, and the fault-filtered GTZAN dataset [43, 44] contains 930 fragments of 30 seconds, both popular for music classification.

Model	Train Dataset	ROC-$\mathrm{AUC}_{\mathrm{TAG}}$	PR-$\mathrm{AUC}_{\mathrm{TAG}}$
CLMR	MSD	**87.8**	**33.1**
CPC	FMA	86.3 (87.8)	30.7 (32.5)
CLMR	FMA	86.2 (86.6)	30.6 (31.2)
CPC	Billboard	85.8 (86.3)	29.7 (30.2)
CPC	GTZAN	83.4 (86.0)	26.9 (29.7)
CLMR	Billboard	82.7 (84.2)	26.9 (27.8)
CLMR	GTZAN	81.9 (85.4)	26.2 (29.5)

Table 3: Transfer learning experiments for CLMR and CPC, which are trained on a separate dataset and evaluated on the MagnaTagATune dataset. The reported scores are obtained with a frozen, pre-trained encoder and a linear classifier. Scores in parenthesis are obtained when adding one extra hidden layer to the classifier.

The results of the transfer learning experiments are shown in Table 3. Both CPC and CLMR show the ability to learn effective representations from out-of-domain datasets without ground truth, and even exceed accuracy scores of previous, supervised end-to-end systems on raw audio [36]. Moreover, both models even demonstrate the ability to learn useful representations on the much smaller GTZAN and Billboard datasets. The CLMR model performs better when it is pre-trained on larger datasets, which is expected as it heavily relies on the number of unique, independent examples that make the contrastive learning task harder, resulting in more robust representations. When pre-training on smaller datasets, CPC can find more useful representations, especially when adding an extra hidden layer to the fine-tune head.

5. CONCLUSION

In this paper, we presented CLMR, a self-supervised contrastive learning framework that learns useful representations of raw waveforms of musical audio. The framework requires no preprocessing of the input audio and is trained without ground truth, which enables simple and straightforward pre-training on music datasets of unprecedented scale. We tested the learned, task-agnostic representations by training a linear classifier on the music classification task on the MagnaTagATune and Million Song datasets, achieving competitive performance compared to fully supervised models. We also showed that CLMR can achieve comparable performance using 100 times fewer labeled songs, and demonstrated the out-of-domain transferability of representations learned from pre-training on entirely different datasets of music. To foster reproducibility and future research on self-supervised learning in music information retrieval, we publicly release the pre-trained models and the source code of all experiments of this paper [2]. The simplicity of training the model without any labels and without preprocessing the audio, together with encouraging results obtained with a single linear layer optimised for a challenging music task, are exciting developments towards unsupervised learning on raw musical audio.

[2] https://github.com/spijkervet/clmr

6. ACKNOWLEDGEMENTS

We would like to thank Jordan B.L. Smith, Wilker Aziz and Keunwoo Choi for their feedback on the draft. We would also like to extend our gratitude to the University of Amsterdam and SURFsara for giving us access to their Research Capacity Computing Services GPU cluster.

7. REFERENCES

[1] F. Korzeniowski and G. Widmer, "A Fully Convolutional Deep Auditory Model for Musical Chord Recognition," *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2016. [Online]. Available: http://arxiv.org/abs/1612.05082

[2] T.-P. Chen and L. Su, "Harmony Transformer: Incorporating Chord Segmentation Into Harmony Recognition," in *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR*, 2019.

[3] F. Korzeniowski and G. Widmer, "End-to-End Musical Key Estimation Using a Convolutional Neural Network," in *25th European Signal Processing Conference (EUSIPCO)*, Kos, Greece, 2017. [Online]. Available: http://arxiv.org/abs/1706.02921

[4] S. Böck, F. Krebs, and G. Widmer, "Joint Beat and Downbeat Tracking with Recurrent Neural Networks," in *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR*, 2016.

[5] J. Pons, O. Nieto, M. Prockup, E. Schmidt, A. Ehmann, and X. Serra, "End-to-End Learning for Music Audio Tagging at Scale," in *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR*, 2017. [Online]. Available: http://arxiv.org/abs/1711.02520

[6] A. van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Advances in Neural Information Processing Systems*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., vol. 26. Curran Associates, Inc., 2013, pp. 2643–2651. [Online]. Available: https://proceedings.neurips.cc/paper/2013/file/b3ba8f1bee1238a2f37603d90b58898d-Paper.pdf

[7] H. V. Koops, W. B. de Haas, J. A. Burgoyne, J. Bransen, A. Kent-Muller, and A. Volk, "Annotator subjectivity in harmony annotations of popular music," *Journal of New Music Research*, vol. 48, no. 3, pp. 232–252, 2019. [Online]. Available: https://doi.org/10.1080/09298215.2019.1613436

[8] P. Hamel, S. Lemieux, Y. Bengio, and D. Eck, "Temporal Pooling and Multiscale Learning for Automatic Annotation and Ranking of Music Audio," in *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR*, 2011, pp. 729–734.

[9] S. Dieleman and B. Schrauwen, "Multiscale Approaches to Music Audio Feature Learning," in *Proceedings of the 14th International Society for Music Information Retrieval conference*, 2013, pp. 116–121.

[10] J. Cramer, H.-H. Wu, J. Salamon, and J. P. Bello, "Look, Listen, and Learn More: Design Choices for Deep Audio Embeddings," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3852–3856.

[11] M. Tagliasacchi, B. Gfeller, F. d. C. Quitry, and D. Roblek, "Pre-Training Audio Representations With Self-Supervision," *IEEE Signal Processing Letters*, vol. 27, pp. 600–604, 2020.

[12] H. Al-Tahan and Y. Mohsenzadeh, "CLAR: Contrastive Learning of Auditory Representations," in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Banerjee and K. Fukumizu, Eds., vol. 130. PMLR, 13–15 Apr 2021, pp. 2530–2538. [Online]. Available: http://proceedings.mlr.press/v130/al-tahan21a.html

[13] A. Saeed, D. Grangier, and N. Zeghidour, "Contrastive Learning of General-Purpose Audio Representations," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 3875–3879.

[14] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative Unsupervised Feature Learning with Exemplar Convolutional Neural Networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 9, pp. 1734–1747, 2015.

[15] A. van den Oord, Y. Li, and O. Vinyals, "Representation Learning with Contrastive Predictive Coding," *arXiv:1807.03748 [cs, stat]*, 2019. [Online]. Available: http://arxiv.org/abs/1807.03748

[16] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," *arXiv:1808.06670 [cs, stat]*, 2019. [Online]. Available: http://arxiv.org/abs/1808.06670

[17] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," *arXiv:2002.05709 [cs, stat]*, 2020, arXiv: 2002.05709. [Online]. Available: http://arxiv.org/abs/2002.05709

[18] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar *et al.*, "Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning," *arXiv preprint arXiv:2006.07733*, 2020.

[19] O. J. Hénaff, A. Razavi, C. Doersch, S. A. Eslami, and A. v. d. Oord, "Data-Efficient Image Recognition with Contrastive Predictive Coding," *arXiv preprint arXiv:1905.09272*, 2019.

[20] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, "Big Self-Supervised Models are Strong Semi-Supervised Learners," *arXiv preprint arXiv:2006.10029*, 2020.

[21] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised Visual Representation Learning by Context Prediction," in *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015, pp. 1422–1430. [Online]. Available: http://ieeexplore.ieee.org/document/7410524/

[22] R. Zhang, P. Isola, and A. A. Efros, "Colorful Image Colorization," in *European conference on computer vision*. Springer, 2016, pp. 649–666.

[23] B. Gfeller, C. Frank, D. Roblek, M. Sharifi, M. Tagliasacchi, and M. Velimirović, "Pitch Estimation Via Self-Supervision," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 3527–3531.

[24] Y. Tian, D. Krishnan, and P. Isola, "Contrastive Multiview Coding," *arXiv preprint arXiv:1906.05849*, 2019.

[25] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum Contrast for Unsupervised Visual Representation Learning," *arXiv preprint arXiv:1911.05722*, 2019.

[26] J. Lee, J. Park, K. L. Kim, and J. Nam, "SampleCNN: End-to-End Deep Convolutional Neural Networks Using Very Small Filters for Music Classification," *Applied Sciences*, vol. 8, no. 1, p. 150, 2018.

[27] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[28] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[29] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, Conference Track Proceedings*, 2016. [Online]. Available: http://arxiv.org/abs/1511.06434

[30] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, "Learning Problem-Agnostic Speech Representations from Multiple Self-Supervised Tasks," in *Proc. Interspeech 2019*, 2019, pp. 161–165. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2605

[31] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-Task Self-Supervised Learning for Robust Speech Recognition," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6989–6993, 2020.

[32] J. W. Kim, J. Salamon, P. Li, and J. P. Bello, "Crepe: A Convolutional Representation for Pitch Estimation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 161–165.

[33] J. Engel, C. Resnick, A. Roberts, S. Dieleman, M. Norouzi, D. Eck, and K. Simonyan, "Neural audio synthesis of musical notes with wavenet autoencoders," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17. JMLR.org, 2017, p. 1068–1077.

[34] E. Kharitonov, M. Rivière, G. Synnaeve, L. Wolf, P.-E. Mazaré, M. Douze, and E. Dupoux, "Data Augmenting Contrastive Learning of Speech Representations in the Time Domain," *arXiv preprint arXiv:2007.00991*, 2020.

[35] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[36] S. Dieleman and B. Schrauwen, "End-to-End Learning for Music Audio," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 6964–6968.

[37] M. Won, K. Choi, and X. Serra, "Semi-supervised Music Tagging Transformer," *In Proc. of International Society for Music Information Retrieval Conference (ISMIR)*, 2021.

[38] E. Law, K. West, M. I. Mandel, M. Bay, and J. S. Downie, "Evaluation of Algorithms Using Games: The Case of Music Tagging," in *Proceedings of the 10th International Society for Music Information Retrieval Conference*, 2009.

[39] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, "The Million Song Dataset," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.

[40] J. Davis and M. Goadrich, "The Relationship between Precision-Recall and ROC Curves," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 233–240. [Online]. Available: https://doi.org/10.1145/1143844.1143874

[41] J. A. Burgoyne, J. Wild, and I. Fujinaga, "An Expert Ground Truth Set for Audio Chord Recognition and Music Analysis," in *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR*, 2011.

[42] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "FMA: A Dataset for Music Analysis," in *18th International Society for Music Information Retrieval Conference, ISMIR*, 2017. [Online]. Available: https://arxiv.org/abs/1612.01840

[43] G. Tzanetakis and P. Cook, "Musical Genre Classification of Audio Signals," *IEEE Transactions on speech and audio processing*, vol. 10, no. 5, pp. 293–302, 2002.

[44] B. L. Sturm, "The GTZAN dataset: Its contents, its faults, their effects on evaluation, and its future use," *arXiv preprint arXiv:1306.1461*, 2013.

[45] J. Spijkervet, "Spijkervet/torchaudio-augmentations," 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5042440

[46] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. R. Zapata, and X. Serra, "ESSENTIA: An Audio Analysis Library for Music Information Retrieval," in *International Society for Music Information Retrieval Conference (ISMIR'13)*, Curitiba, Brazil, 04/11/2013 2013, pp. 493–498. [Online]. Available: http://hdl.handle.net/10230/32252

[47] U. Zölzer, X. Amatriain, D. Arfib, J. Bonada, G. De Poli, P. Dutilleux, G. Evangelista, F. Keiler, A. Loscos, D. Rocchesso *et al.*, *DAFX-Digital Audio Effects*. John Wiley & Sons, 2002.

[48] M. R. Schroeder, "Natural Sounding Artificial Reverberation," *Journal of the Audio Engineering Society*, vol. 10, no. 3, pp. 219–223, July 1962.

[49] L. v. d. Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of machine learning research*, vol. 9, pp. 2579–2605, 2008.

[50] S. Stevens, J. Volkmann, and E. B. Newman, "A Scale for the Measurement of the Psychological Magnitude Pitch," *Journal of the Acoustical Society of America*, vol. 8, pp. 185–190, 1937.

[51] E. Zwicker, "Subdivision of the Audible Frequency Range into Critical Bands (Frequenzgruppen)," *Acoustical Society of America Journal*, vol. 33, no. 2, p. 248, Jan. 1961.