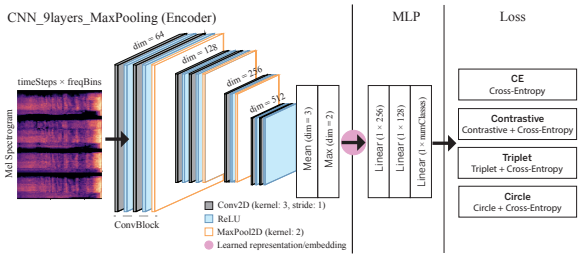

Figure 1. Network architecture: CNN9-max encoder, MLP classifier head, and loss functions.

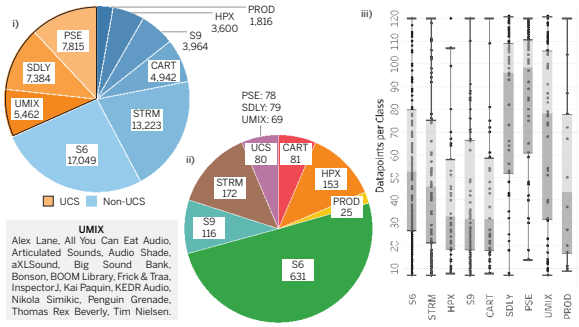

Figure 2. Imbalanced UCS & Non-UCS dataset statistics, (i) Number of audio-files per training dataset, (ii) Number of classes per taxonomy, (iii) Class distribution sorted from most to least classes among training datasets.

3. METHOD

Our best pre-trained model is freely available online. [3]

3.1 Model Architecture

Our encoder architecture adapts the Convolutional Neural Network CNN9-max architecture from a DCASE 2019 baseline system for multi-class classification and is displayed in Fig. 1 [40]. We pre-train the encoder with one Multi-Layer Perceptron (MLP) classifier head per dataset. After training, we freeze the encoder and pass the representations to a simple Nearest Neighbor classifier to train and test the learned embedding space. Models were trained with Python, PyTorch, and the PyTorch metric learning library. [4]

3.2 Input Representation

All audio files were re-sampled to 44.1 kHz, down-mixed, and normalized. Mel-spectrograms were computed with a block size of 46 ms, hop size of 23 ms, 96 Mel-Bins, and span the audible frequency range [41]. We used min-max normalization for spectrograms and z-score standardization for extracted embeddings. Spectrogram input dimensions to the network are (100, 96) and span approximately 2 s of audio. This length was determined in pilot experiments.

3.3 Training Procedure

We trained all models with a batch size of 64, the Adam optimizer, and early stopping. Model checkpoints used for inference correspond to the best validation loss. For cross-dataset training, we save the best average validation loss from all datasets. Hyper-parameters for pre-training were found via 20 trials of random search. During pre-training, 2 s frames of audio files are randomly sampled whenever a batch is retrieved. We re-shuffle the dataset per epoch. We compute embeddings with 50% overlap at inference time.

3.4 Evaluation Metrics

The macro F-1 score over classes in a dataset is our predominant metric for evaluating classification performance, as all datasets have imbalanced class distributions. We also monitor the Davies-Bouldin Index (DBI) to evaluate the quality of the embedding space structure, i.e., clustering. The DBI measures the average similarity between a cluster and its most similar cluster.

[3] github.com/alisonbma/aiSFX, last accessed August 11, 2022.

[4] kevinmusgrave.github.io/pytorch-metric-learning, last accessed May 10, 2022.

4. DATASETS & TAXONOMIES

We address our goal of training generalized embeddings by training and testing with 9 datasets and 7 taxonomies. We emphasize that we use a diverse assortment of datasets for this research and present corresponding data statistics in Fig. 2. We use 3 UCS-compliant datasets: Pro Sound Effects (*PSE*), Soundly (*SDLY*), UCS Mixed (*UMIX*), and 6 Non-UCS datasets from the Sound Ideas library: Cartoon Express (*CART*), HPX Digital (*HPX*), Production Elements (*PROD*), Series 6000 (*S6*), Series 9000 (*S9*), Soundstorm (*STRM*). For UCS-compliant datasets, we use UCSv8.1 *Category* labels as ground truth.

Preliminary experiments with UCS lead us to use experimental subsets of 150 or fewer datapoints per class. We pre-train and evaluate embeddings on the full imbalanced subsets (65k audio files) except in *Cross-Dataset* training scenarios, where we use class-balanced versions for Non-UCS data. A stratified train validation test split of 8:1:1 was conducted. For UCS-compliant datasets, the fine *CatID* labels determined stratification.

We select *PSE* as the base dataset for *UCS-Transfer* experiments because of its larger number of datapoints and classes, cleaner labels compared to Non-UCS datasets, better generalization when evaluated against other UCS datasets, and more even distribution among both levels of the UCS 2-level class hierarchy.

5. EXPERIMENTAL SETUP

We list our research questions and hypotheses below using the following experimental variables:

- *Metric learning loss functions (Fig. 1, Sect. 5.1)*,
- *Variable training scenarios (Sect. 5.2)*, and
- *Cross-dataset training methods* (Sect. 5.3).

5.1 RQ1: Impact of Metric Learning

We first ask whether metric learning loss functions that learn distances instead of absolute class labels will improve

classification results. We hypothesize that metric learning approaches will learn a more discriminative embedding space compared to CE models and that this better-structured embedding space will improve classification results.

5.1.1 Training Parameterization

We train all metric learning models with Khosla's Joint-Training scheme that optimizes a hybrid sum of a metric learning loss function and Cross-Entropy loss as it has been shown to be more effective than a Two-Step training method [26, 33, 35, 42, 43]. Fig. 1 illustrates our experiments with the joint metric learning losses (i) Contrastive, (ii) Triplet, and (iii) Circle.

The metric learning losses optimize cosine similarity. Positive and negative samples are computed from all possible pairs or triplets between samples in a batch. Accordingly, we find that our batch size is sufficient for training from initial experiments. Model training utilized online mining and the *regularface* embedding regularizer [44].

In addition to re-shuffling the dataset per epoch and randomly selecting 2 s frames, we re-sample the dataset so that it randomly selects new datapoints per class per epoch. Here, we equally sample 4 datapoints per class so that a single batch includes 16 different classes; we found minimal difference from weighted sampling.

5.2 RQ2: Impact of Cross-Dataset Training

Secondly, we ask whether cross-dataset training will improve our best Cross-Entropy and metric learning results in all training scenarios. We hypothesize that a *Cross-Dataset* training scenario will outperform all others, including *UCS-Transfer* (see definition below), improving upon both UCS & Non-UCS dataset results. Our rationale is that representations generated from *Cross-Dataset* models will have seen a more significant quantity and variety of data from different sound taxonomies, producing more generalized representations. Moreover, we hypothesize that *UCS-Transfer* will yield decent classification results as UCS is a standardized taxonomy for sound effects libraries. We denote *Cross-Dataset* models with the prefix, *X*.

5.2.1 Training Scenario Definitions

We conduct experiments with 3 training scenarios: (i) *Within-Dataset* training: Pre-train and evaluate the encoder on the same dataset, (ii) *UCS-Transfer*: Pre-train the encoder on a UCS-compliant dataset and evaluate on other datasets in a transfer learning scenario, and (iii) *Cross-Dataset* training: Pre-train the encoder on all datasets and taxonomies, evaluate the encoder on any dataset.

5.3 RQ3: Impact of Cross-Dataset Training Methods

Datasets used for training may have varying characteristics. For example, they may use taxonomies of dissimilar scopes and biases and have different dataset sizes, all of which an effective classifier must adapt to. Our final question is whether training scenarios accommodating dataset characteristics will improve classification results. We experiment with 3 methods as introduced in the following subsections, (i) Data mixing (*X-Sequential*, *X-Joint*), (ii) 'Focal' dataset regularization (*FDR*), and (iii) Dataset-independent BatchNorm layers (*BN*).

5.3.1 Data Mixing

We use two variations of data mixing. *Sequential* trains on all datapoints from a single dataset before training on the next, i.e., concatenate datasets, while *Joint* trains on datapoints from all datasets in mixed order, similar to stratified sampling in the literature [36]. We limit mixing for *Joint* so that all datapoints in a batch must correspond to the same dataset. We affirm that backpropagation is called per batch.

5.3.2 'Focal' Dataset Regularization

We regularize the datasets by 'difficulty' and reweigh them by a dataset's training convergence speed in epochs. Inspired by Focal Loss [38, 45, 46] and mining for hard pairs or triplets in metric learning [29, 47–49], easier datasets are down-weighted so that training focuses on difficult datasets.

$$\alpha_d = \frac{1 - \beta^{n_e}}{1 - \beta} \tag{1}$$

We modify the reweighting function shown in Eqn. 1 to re-balance the datasets and initialize these weights with n_e, the epoch at which a dataset's training set macro F-1 score crosses a threshold of 90% [50]. These weights may be adjusted with hyperparameter β to reduce or heighten the difference between dataset weights. a_d is the unnormalized reweighing factor per dataset, d. As previously done in the literature, we normalize α_d so that $\sum_{d=1}^{D} \alpha_d = D$, keeping the loss in roughly the same range. D represents the total number of datasets used for pre-training [45, 50].

5.3.3 Dataset-Independent BatchNorm Layers

Similar to selecting the correct classifier head per dataset when pre-training the encoder, we select a dataset's corresponding BatchNorm layers. This is equivalent to turning each ConvBlock in Fig. 1 into a Dataset-Aware Block [39].

5.4 RQ4: Comparison to SoTA

We select OpenL3 features as our reference to SoTA deep audio embeddings because L^3-Net is said to "consistently outperform VGGish and SoundNet on environmental sound classification" [18]. We use the default OpenL3 parameters as they yield the highest frame-level classification results and select the *Music* content type embeddings for comparison (6144 dimensionality, 0.1 s hop size, and 256 Mel Bins) [18]. We aggregate these features to represent 2 s of an audio file for consistency with our input representation.

6. RESULTS

All results are computed at the frame-level, i.e., aggregated over each 2 s frame extracted from audio files in the hold-out test set. Boxplot datapoints indicate results for a single dataset. Metric learning plots, Fig. 4 and 5, only show *X-Sequential* results for the *Cross-Dataset* training scenario.

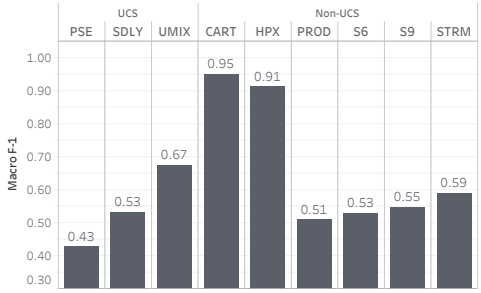

Figure 3. Baseline macro F-1 score classification results using Cross-Entropy loss, CE (*Within-Dataset*).

6.1 Baseline Results

Fig. 3 displays baseline classification results where each model is trained and tested on only one dataset, itself. Plots for the following experiments only show the change from this baseline plot.

6.2 RQ1: Impact of Metric Learning

Fig. 4 shows how metric learning models improve upon the baseline's embedding space structure to different degrees. A lower DBI indicates a better embedding space structure. Contrastive improves the space for most datasets in all training scenarios. Triplet results in a slightly higher average DBI in all but the *UCS-Transfer* training scenario. Circle performs significantly better in a *Within-Dataset* training scenario with an average DBI decrease of 0.43. However, it performs surprisingly poorly in *UCS-Transfer* & *Cross-Dataset*, suggesting that it is not ideal for generalization, a prime motivator of representation learning.

Contrary to our hypothesis, a better-structured embedding space (lower DBI) does not always improve classification results (higher macro F-1), shown in Fig. 5. We only see that metric learning models improve classification results in the *UCS-Transfer* training scenario. We specifically note that Circle significantly improves the embedding space structure compared to Contrastive in *Within-Dataset*. However, both embedding spaces still yield similar classification performance, an average of -1.75% and -1.50% from CE, respectively. We note other work by Lee et al. regarding the similarities between metric learning and classification [51].

Focusing on classification results, we identify Triplet as our 'best' metric learning loss with a slightly higher average performance than Contrastive. Triplet performs on par with CE in *Within-Dataset* and *Cross-Dataset* (less than 1% difference on average), and has an average of 4.25% improvement from CE in *UCS-Transfer*. Therefore, Contrastive and Circle loss results will be omitted in the following sections.

6.3 RQ2: Impact of Cross-Dataset Training

Results for cross-dataset training are shown in Fig. 6. Matching our hypothesis, we find that for CE models, *Cross-Dataset* outperforms *Within-Dataset* models for the majority of datasets by approximately 4-5%. We find that Triplet models also tend to improve by 1-3% except for *X-Joint-BN + Triplet*. This confirms that cross-dataset training indeed leads to better generalization across all sound taxonomies.

6.3.1 UCS Insights

We preface this section with preliminary results on UCS. We find that the UCS-compliant datasets in this study use the standardized UCS taxonomy in a non-uniform way. Without fine-tuning another classifier head, one cannot use a UCS taxonomy-trained model on other UCS-compliant datasets. This corroborates the need for generalized representations that can adapt to any taxonomy of sound.

With this said, Fig. 6 also shows that *UCS-Transfer* can achieve results considerably better than random. Although it under-performs *Within-Dataset* training by an average of 7.48% with CE models, we find that there is only an average of -1.62% difference from CE with Triplet models. Alongside, we also verify that other UCS-compliant libraries, *SDLY* and *UMIX*, can be used as base sets to achieve decent generalization in a transfer learning scenario.

We note some dataset-specific details for *UCS-Transfer + Triplet* compared to our best *Cross-Dataset* Triplet model, *X-Joint + Triplet*. Both perform similarly (around 1% worse) on Non-UCS *CART, HPX, PROD*, and *STRM* datasets. *UCS-Transfer* performs around 5-6% worse on UCS-compliant *SDLY* and *UMIX*, a substantial improvement from previous results that did not re-train a classifier head. Lastly, it performs around 7% worse on Non-UCS *S6*, likely because *S6* exhibits the highest number of classes amongst all datasets while *PSE* only has 78 classes.

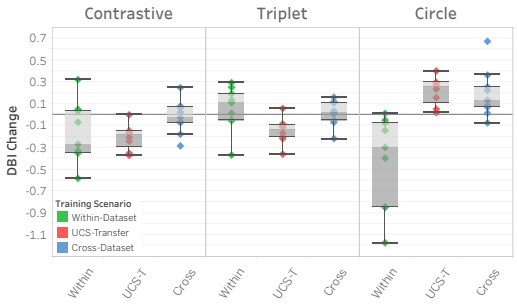

Figure 4. DBI change of metric learning models relative to CE in different training scenarios.

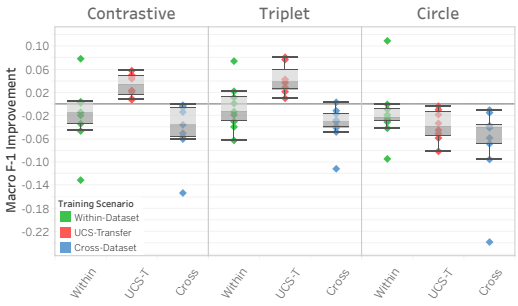

Figure 5. Macro F-1 score improvement of metric learning models relative to CE in different training scenarios.

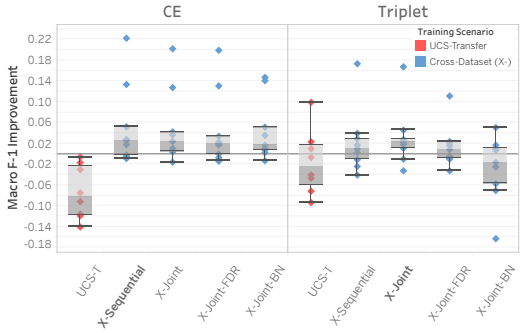

Figure 6. Macro F-1 score improvement of various training scenarios from *Within-Dataset*.

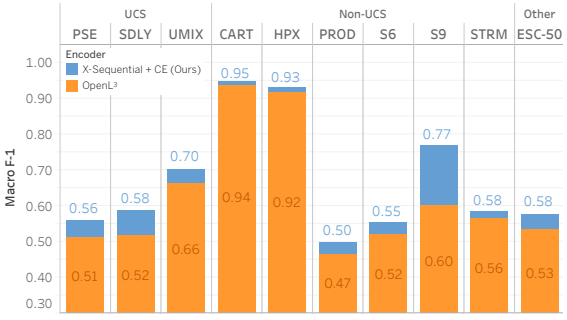

Figure 7. Our best model compared to OpenL3 audio embeddings. Blue bars illustrate the amount in which we outperform OpenL3.

6.4 RQ3: Impact of Cross-Dataset Training Methods

We experiment to see if accommodating different dataset characteristics with 3 cross-dataset training methods will further improve results. We compare all cross-dataset training methods to *X-Sequential* in our discussion.

First, sensitivity to encoder bias and training order does not severely impact results for our task. Data mixing results note an average 1% difference between *X-Sequential* and *X-Joint* in CE and Triplet.

Second, reweighing datasets as proposed in Sect. 5.3.2 does not lead to observable consistent trends. We show results that use $\beta = 0.999$. Looking at difficult datasets, the performance of *SDLY* worsens by 2.96%, while *STRM* improves by 3.27% despite having similar convergence speeds and being the second-ranked most 'difficult' datasets. We also observe an unfortunate worsening of results for faster-converging datasets, i.e., *HPX* decreases by 1.77% (Triplet), *S9* decreases by 2.32% (CE) and 6.20% (Triplet).

Third, we verify that independent BatchNorm layers do preserve dataset-specific characteristics. Unlike other work in the literature [39], we find that this may not always be beneficial as it may exacerbate dataset-specific problems. Shown in Fig. 6, we specifically note decreases in performance for Non-UCS *HPX, PROD, S6*, and *S9*, which we attribute to messier inconsistent labels and insufficient training data, apparent in Fig. 2. This decrease can be quite dramatic with *S9* dropping by 7.53% (CE) and 24.4% (Triplet). Alongside, we find that UCS-compliant *PSE* and *SDLY*, as well as Non-UCS *STRM* experience a 2-3$\%$ increase in performance, which we attribute to both the cleaner UCS taxonomy and datasets having sufficient training data.

6.5 RQ4: Comparison to SoTA

Overall, our work reveals that our best model is *X-Sequential + CE*. We compare *X-Sequential + CE* and OpenL3 in Fig. 7 and provide a commonly used benchmark dataset, ESC-50 [52], to extend the assessment of our work in a non-task-specific audio application. Unlike the rest of our datasets, we show the 5-fold cross-validation results for ESC-50 as commonly reported in the literature. Ultimately, we observe that our best model outperforms OpenL3 on all datasets. In addition, we note that our model requires fewer resources to achieve its results. While both models have a similar number of parameters (4.7M), our best model is trained on approximately 7x less data (39k audio-files vs. 296k videos) [18].

7. CONCLUSION

We introduce a new pre-trained representation for automatic sound effects library classification. Our task-specific representation outperforms OpenL3 on both UCS & Non-UCS taxonomies across diverse datasets. Moreover, we show the effectiveness of cross-dataset training with Cross-Entropy loss over metric learning for this task. Results suggest that the quality and diversity of datasets are key to pre-training robust representations, supportive of preliminary experimental results where pre-training on increasing quantities of similar data yielded improvements with diminishing returns. Our contributions include that we are the first to conduct extensive deep learning experiments on UCS, experiment on relevant data for a current problem, and investigate an under-researched aspect of representation learning with cross-dataset training in the audio domain.

We believe cross-dataset training is a prerequisite for future work. Our current research does not leverage all datasets available to us nor the abundance of taxonomies they are comprised of given that many datasets did not have sufficient labels for training. To leverage these resources, methods to effectively work with large amounts of diverse data become increasingly essential. Selecting optimal cross-dataset training methods is fundamental before introducing new techniques, such as semi-supervised or self-supervised learning to work with datasets of varying label quality [53,54]. Accordingly, we would like to (i) look into the connection between multi-task learning and cross-dataset training to understand the specific advantages and drawbacks of these approaches and (ii) conduct experiments to assess the impact of increasing the amount of training data, observing the relationship of this with more complex network architectures and other pre-training methods [55]. Finally, we hope to deepen our understanding of UCS by investigating the non-uniformity of class labels between UCS-compliant datasets in this study, looking at the interpretability and disentanglement of our representations [51], and exploring the concept of a generalized embedding from the perspective of taxonomy conversion [56].

8. REFERENCES

[1] J. Yang, Y. Zhang, and Y. Hai, "Retrieval and management system for layer sound effect library," *Cognitive Computation and Systems*, vol. 2, no. 4, pp. 247–253, 2020.

[2] G. Lafay, N. Misdariis, M. Lagrange, and M. Rossignol, "Semantic browsing of sound databases without keywords," *Journal of the Audio Engineering Society*, September 2016.

[3] S. Säger, B. Elizalde, D. Borth, C. Schulze, B. Raj, and I. Lane, "Audiopairbank: towards a large-scale tag-pair-based audio content analysis," *EURASIP Journal on Audio, Speech, and Music Processing*, p. 12, 2018.

[4] Y. Zhang, J. Hu, Y. Zhang, B. Pardo, and Z. Duan, "Vroom!: A search engine for sounds by vocal imitation queries," in *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval (CHIIR)*. Association for Computing Machinery, 2020, pp. 23–32.

[5] G. G. Peeters and J. D. Reiss, "A deep learning approach to sound classification for film audio post-production," *Journal of the Audio Engineering Society*, May 2020.

[6] E. Fonseca, M. Plakal, F. Font, D. P. W. Ellis, X. Favory, J. Pons, and X. Serra, "General-purpose tagging of freesound audio with audioset labels: Task description, dataset, and baseline," *arXiv preprint arXiv:1807.09902*, 2018.

[7] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 467–476, 2008.

[8] X. Favory, F. Font, and X. Serra, "Search result clustering in collaborative sound collections," in *Proceedings of the 2020 International Conference on Multimedia Retrieval (ICMR '20)*, 2020, pp. 207–214.

[9] F. Font, G. Roma, and X. Serra, *Sound Sharing and Retrieval*. Cham: Springer International Publishing, 2018, pp. 279–301.

[10] D. Moffat, D. Ronan, and J. D. Reiss, "Unsupervised taxonomy of sound effects," in *Proceedings of the 20th International Conference on Digital Audio Effects (DAFx)*, 2017.

[11] G. Lemaitre and L. M. Heller, "Evidence for a basic level in a taxonomy of everyday action sounds," *Experimental Brain Research*, vol. 226, no. 2, pp. 253–264, 2013.

[12] O. Bones, T. J. Cox, and W. J. Davies, "Sound categories: Category formation and evidence-based taxonomies," *Frontiers in Psychology*, vol. 9, pp. 1664–1078, 2018.

[13] B. Elizalde, R. Revutchi, S. Das, B. Raj, I. Lane, and L. M. Heller, "Identifying actions for sound event classification," in *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2021, pp. 26–30.

[14] B. M. Elizalde, "Never-ending learning of sounds," Ph.D. dissertation, Carnegie Mellon University, 2020.

[15] F. Font, G. Roma, and X. Serra, "Freesound technical demo," in *Proceedings of the 21st ACM International Conference on Multimedia (MM)*. Association for Computing Machinery, 2013, pp. 411–412.

[16] A. Pearce, T. Brookes, and R. Mason, "Timbral attributes for sound effect library searching," *Journal of the Audio Engineering Society*, June 2017.

[17] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2017, pp. 131–135.

[18] J. Cramer, H.-H. Wu, J. Salamon, and J. P. Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2019, pp. 3852–3856.

[19] R. Arandjelovic and A. Zisserman, "Look, listen and learn," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017, pp. 609–617.

[20] Y. Wang, N. J. Bryan, M. Cartwright, J. P. Bello, and J. Salamon, "Few-shot continual learning for audio classification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021, pp. 321–325.

[21] Y. Wang, N. Bryan, J. Salamon, M. Cartwright, and J. Bello, "Who calls the shots? rethinking few-shot learning for audio," in *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2021, pp. 36–40.

[22] S. Gidaris and N. Komodakis, "Dynamic few-shot visual learning without forgetting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4367–4375.

[23] H. F. Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging hierarchical structures for few-shot musical instrument recognition," in *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021, pp. 220–228.

[24] E. Fonseca, D. Ortego, K. McGuinness, N. E. O'Connor, and X. Serra, "Unsupervised contrastive learning of sound event representations," in *Proceedings of the*

IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2021, pp. 371–375.

[25] J. Spijkervet and J. A. Burgoyne, "Contrastive learning of musical representations," in *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021, pp. 673–681.

[26] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 18 661–18 673.

[27] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *International Workshop on Similarity-Based Pattern Recognition*. Springer International Publishing, 2015, pp. 84–92.

[28] K. Q. Weinberger, J. Blitzer, and L. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Advances in Neural Information Processing Systems (NeurIPS)*, Y. Weiss, B. Schölkopf, and J. Platt, Eds., vol. 18. MIT Press, 2005.

[29] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.

[30] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 6398–6407.

[31] R. Zhang, H. Wu, W. Li, D. Jiang, W. Zou, and X. Li, "Transformer based unsupervised pre-training for acoustic representation learning," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021, pp. 6933–6937.

[32] A. Saeed, D. Grangier, and N. Zeghidour, "Contrastive learning of general-purpose audio representations," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021, pp. 3875–3879.

[33] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "Byol for audio: Exploring pre-trained general-purpose audio representations," *arXiv preprint arXiv:2204.07402*, 2022.

[34] L. Wang, P. Luc, Y. Wu, A. Recasens, L. Smaira, A. Brock, A. Jaegle, J. Alayrac, S. Dieleman, J. Carreira, and A. van den Oord, "Towards learning universal audio representations," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2022, pp. 4593–4597.

[35] P. Seshadri and A. Lerch, "Improving music performance assessment with contrastive learning," in *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021, pp. 634–641.

[36] H.-W. Dong, K. Chen, J. McAuley, and T. Berg-Kirkpatrick, "Muspy: A toolkit for symbolic music generation," in *Proceedings of the 21st International Society for Music Information Retrieval Conference (ISMIR)*, 2020, pp. 101–108.

[37] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, pp. 1623–1637, 2022.

[38] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018, pp. 4879–4883.

[39] L. Wang, D. Li, Y. Zhu, L. Tian, and Y. Shan, "Cross-dataset collaborative learning for semantic segmentation in autonomous driving," *arXiv preprint arXiv:2103.11351*, 2021.

[40] Q. Kong, Y. Cao, T. Iqbal, Y. Xu, W. Wang, and M. D. Plumbley, "Cross-task learning for audio tagging, sound event detection and spatial localization: DCASE 2019 baseline systems," *arXiv preprint arXiv:1904.03476*, 2019.

[41] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. Zapata, and X. Serra, "Essentia: an audio analysis library for music information retrieval," in *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, 2013, pp. 493–498.

[42] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 18 661–18 673.

[43] J. Fan, E. Nichols, D. Tompkins, A. E. M. Méndez, B. Elizalde, and P. Pasquier, "Multi-label sound event retrieval using a deep learning-based siamese structure with a pairwise presence matrix," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2020, pp. 3482–3486.

[44] K. Zhao, J. Xu, and M.-M. Cheng, "Regularface: Deep face recognition via exclusive regularization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1136–1144.

[45] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017, pp. 2999–3007.

[46] K. Nada, K. Imoto, R. Iwamae, and T. Tsuchiya, "Multitask learning of acoustic scenes and events using dynamic weight adaptation based on multi-focal loss," in *Proceedings of the 13th Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2021, pp. 1156–1160.

[47] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krahenbuhl, "Sampling matters in deep embedding learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017, pp. 2840–2848.

[48] H. Xuan, A. Stylianou, and R. Pless, "Improved embeddings with easy positive triplet mining," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2020, pp. 2474–2482.

[49] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott, "Multi-similarity loss with general pair weighting for deep metric learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 5022–5030.

[50] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 9260–9269.

[51] J. Lee, N. Bryan, J. Salamon, Z. Jin, and J. Nam, "Metric learning vs classification for disentangled music representation learning," in *Proceedings of the 21st International Society for Music Information Retrieval Conference (ISMIR)*, 2020, pp. 439–445.

[52] K. J. Piczak, "Esc: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM International Conference on Multimedia (MM)*. New York, NY, USA: Association for Computing Machinery, 2015, pp. 1015–1018.

[53] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 596–608.

[54] S. Gururani and A. Lerch, "Semi-supervised audio classification with partially labeled data," in *2021 IEEE International Symposium on Multimedia (ISM)*, 2021, pp. 111–114.

[55] T. Chen, Y. Xie, S. Zhang, S. Huang, H. Zhou, and J. Li, "Learning music sequence representation from text supervision," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2022, pp. 4583–4587.

[56] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mT5: A massively multilingual pre-trained text-to-text transformer," in *2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*. "Association for Computational Linguistics", 2021, pp. 483–498.

9. ACKNOWLEDGMENTS

We would like to thank those who provided the data required to conduct this research as well as those who took the time to share their insights and software licenses for tools regarding sound search, query, and retrieval.

Alex Lane [5]

All You Can Eat Audio [6] [7]

Articulated Sounds [8]

Audio Shade [9] [10]

aXLSound [11]

Big Sound Bank [12]

BaseHead [13]

Bonson [14] [15]

BOOM Library [16]

Frick & Traa [17]

[5] Alex Lane, *Resonant Rusty Door*. [Dataset]. Available: https://www.alex-lane.com. [Accessed: September 15, 2021].

[6] All You Can Eat Audio, *It's A Plain Phone*. [Dataset]. Available: https://allyoucaneataudio.com/library-packs/it-is-a-plain-phone-p04. [Accessed: September 15, 2021].

[7] All You Can Eat Audio, *Cuff 'Em*. [Dataset]. Available: https://allyoucaneataudio.com/library-packs/cuff-em. [Accessed: September 15, 2021].

[8] Articulated Sounds, *Starter Pack*. [Dataset] Available: https://articulatedsounds.com/audio-royalty-free-library/sfx/free-global-sample-pack. [Accessed: September 15, 2021].

[9] Audio Shade, *Bravo Dramatic Audiences*. [Dataset]. Available: https://audioshade.com/shop. [Accessed: September 15, 2021].

[10] Audio Shade, *Brutality - Gore and Combat FX Toolkit*. [Dataset]. Available: https://audioshade.com/shop. [Accessed: September 15, 2021].

[11] aXLSound, *Planes - Takeoffs & Landings*. [Dataset]. Available: https://axlsound.com. [Accessed: September 15, 2021].

[12] Big Sound Bank, *Big Sound Bank*. [Dataset]. Available: https://bigsoundbank.com. [Accessed: September 15, 2021].

[13] baseheadinc.com, last accessed September 15, 2021.

[14] Bonson, *Wings*. [Dataset]. Available: https://www.bonson.ca/product/wings. [Accessed: September 15, 2021].

[15] Bonson, *Moves*. [Dataset]. Available: https://www.bonson.ca/product/moves. [Accessed: September 15, 2021].

[16] BOOM Library, *Cyber Weapons*. [Dataset]. Available: https://www.boomlibrary.com/sound-effects/cyber-weapons. [Accessed: September 15, 2021].

[17] Frick & Traa, *City Bicycles*. [Dataset]. Available: https://www.frickandtraa.com/webshop/sound-libraries. [Accessed: September 15, 2021].

Hzandbits [18]

InspectorJ [19]

Kai Paquin [20]

KEDR Audio [21] [22] [23] [24] [25]

Krotos Audio [26] [27]

Nikola Simikic [28]

Penguin Grenade [29] [30] [31]

Pro Sound Effects [32]

Rick Allen Creative [33]

Sononym [34]

Sound Ideas [35]

Soundly [36]

Soundminer [37]

Storyblocks [38]

Tim Nielsen [39] [40] [41] [42]

Thomas Rex Beverly [43] [44]

ZapSplat [45]

[18] C. Hagelskjaer, "Interview with Christian Hagelskjaer," July 28, 2021.

[19] InspectorJ, *96 General Library*. [Dataset]. Available: https://inspectorj.sellfy.store/p/96-general-library-bundle. [Accessed: September 15, 2021].

[20] K. Paquin, *Kailibrary*. [Dataset].

[21] KEDR Audio, *Cinemaphone*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/cinemaphone-ringtones-notifications. [Accessed: September 15, 2021].

[22] KEDR Audio, *Kinetics: Construction Kit*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/kinetics-construction-kit. [Accessed: September 15, 2021].

[23] KEDR Audio, *Kinetics: Cinematic Tension*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/kinetics-cinematic-tension. [Accessed: September 15, 2021].

[24] KEDR Audio, *Radio Nomad*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/radio-nomad. [Accessed: September 15, 2021].

[25] KEDR Audio, *Vibrations*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/vibrations. [Accessed: September 15, 2021].

[26] Krotos Audio, *Ammo and Reloads*. [Dataset]. Available: https://www.krotosaudio.com/products/ammo-reloads-sound-effects-library. [Accessed: September 15, 2021].

[27] Krotos Audio, *Mechanical*, 1. [Dataset]. Available: https://www.krotosaudio.com/products/mechanical-sound-effects-library-vol-1. [Accessed: September 15, 2021].

[28] N. Simikic, *Nikola Simikic Sound Library*. [Dataset].

[29] Penguin Grenade, *Arcs and Sparks*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/arcs-and-sparks. [Accessed: September 15, 2021].

[30] Penguin Grenade, *Explosive Energy*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/explosive-energy. [Accessed: September 15, 2021].

[31] Penguin Grenade, *Holograms*. [Dataset]. Available: https://www.asoundeffect.com/sound-library/holograms. [Accessed: September 15, 2021].

[32] Pro Sound Effects, *CORE 2*, Pro. [Dataset]. Available: https://www.prosoundeffects.com/core-2. [Accessed: September 15, 2021].

[33] R. Allen, "Interview with Rick Allen Creative," August 06, 2021.

[34] B. Næsby, "Interview with Bjørn Næsby from Sononym," September 10, 2021.

[35] Sound Ideas, *Sound Ideas Library*. [Dataset]. Available: https://www.sound-ideas.com. [Accessed: February 01, 2022].

[36] Soundly, *Soundly*. [Dataset]. Available: https://www.soundly.com. [Accessed: February 01, 2022].

[37] store.soundminer.com, last accessed September 15, 2021.

[38] storyblocks.com, last accessed September 15, 2021.

[39] T. Nielsen, *Early Mother Communicator*. [Dataset].

[40] T. Nielsen, *Ether*. [Dataset].

[41] T. Nielsen, *Yellowstone*. [Dataset].

[42] T. Nielsen, *Baltic*. [Dataset].

[43] T. R. Beverly, *Maine Bundle*. [Dataset]. Available: https://thomasrexbeverly.com/collections/sound-libraries/products/maine-bundle. [Accessed: September 15, 2021].

[44] T. R. Beverly, *Wild Animal Calls*. [Dataset]. Available: https://thomasrexbeverly.com/products/wild-animal-calls. [Accessed: September 15, 2021].

[45] A. McKinny, "Interview with Alan McKinny from ZapSplat," July 30, 2021.

CONCEPT-BASED TECHNIQUES FOR "MUSICOLOGIST-FRIENDLY" EXPLANATIONS IN A DEEP MUSIC CLASSIFIER

Francesco Foscarin[1*] **Katharina Hoedt**[1*] **Verena Praher**[1*]
Arthur Flexer[1] **Gerhard Widmer**[1,2]
[1] Institute of Computational Perception, Johannes Kepler University Linz, Austria
[2] LIT AI Lab, Linz Institute of Technology, Austria
{firstname}.{lastname}@jku.at

ABSTRACT

Current approaches for explaining deep learning systems applied to musical data provide results in a low-level feature space, e.g., by highlighting potentially relevant time-frequency bins in a spectrogram or time-pitch bins in a piano roll. This can be difficult to understand, particularly for musicologists without technical knowledge. To address this issue, we focus on more human-friendly explanations based on high-level musical concepts. Our research targets trained systems (post-hoc explanations) and explores two approaches: a supervised one, where the user can define a musical concept and test if it is relevant to the system; and an unsupervised one, where musical excerpts containing relevant concepts are automatically selected and given to the user for interpretation. We demonstrate both techniques on an existing symbolic composer classification system, showcase their potential, and highlight their intrinsic limitations.

1. INTRODUCTION

The mass adoption of deep learning methods in recent years has increased interest in the field of *explainability*, [1] i.e., the study of techniques that generate a human-understandable explanation of a model's decision [1]. As deep learning models are usually not intrinsically interpretable, techniques that can be applied to trained models (i.e., *post-hoc* methods) are of great interest. The resulting explanations cannot only reveal potential issues of the system itself and the data it uses, but can also provide insights into the problem we are targeting, thus helping us to gain knowledge about it [2].

Higher-level musical tasks such as chord transcription or composer classification may require explanations that can only be understood by persons with advanced musical expertise. However, the explanation techniques for musical systems that have been proposed in recent years [3–7] are *feature-based*, i.e., the explanation is given in terms of the input features the system considers. Typical input features for musical systems, e.g., spectrograms or piano roll representations, are high-dimensional, and, since the importance of a single time frequency / pitch bin does not convey much meaningful interpretation, feature-based explanations can be hard to understand. [2] Musicologists are therefore often unable to analyse the results of trained systems, let alone contribute to the development of new learning models. This motivates research on techniques that provide explanations that are as similar as possible to those that a human music domain expert would naturally use.

Concept-based explanations offer an interesting direction. They were first explored by Kim et al. [12] and later developed in several works (e.g., by Chen et al. [13]) for image systems, which also use high-dimensional input features. Instead of producing feature-level descriptors, the explanation is based on human-understandable concepts. For example, [12] tests whether the concept of "stripes" would increase the probability that an image classifier labels an image as "zebra". Music can also be described with *musical concepts*; terms such as "diatonic sequence", "alberti bass", "difficult-to-play music", "orchestral music", "rubato", "shuffle drum beat", "funky bass line", etc. are used to describe pieces or specific elements in a piece.

In this paper, we explore two concept-based techniques to explain deep learning systems that deal with musical data: one supervised and one unsupervised. The first (see Section 4) is based on Testing with Concept Activation Vectors (TCAV) [12]: the user defines a concept by providing examples and interrogates the system to find out if the concept is relevant or not for its decision. This can be applied to any kind of neural network and, even more generally, to any system that has a hidden layer for which we can compute directional derivatives. The second technique, described in Section 5, is an adaptation of [14] and works in an unsupervised fashion, where the most relevant concepts are automatically produced and given to the user for interpretation. Each concept is presented in the form of a set of musical excerpts. This approach requires networks whose hidden layers contain only non-negative values and

* Equal contribution.
[1] Considered synonymous to the term *interpretability* in this paper.

[2] Moreover, their truthfulness has recently been debated [8–11].

have a spatial correlation with the input data, two conditions that are satisfied by most Convolutional Neural Networks (CNNs).

Our contributions are: the first application of concept-based post-hoc approaches to musical data, in particular, we target the composer classification system of Kim et al. [15] that uses piano roll representations of piano MIDI files as input; the definition and creation of musical concept datasets; a dedicated visualisation of unsupervised concepts for symbolic music data; and, finally, the exploration of the Non-negative Tucker Decomposition (NTD) for the factorisation of hidden layers. Our code and data are available on Github. [3]

2. RELATED WORK

Recent work in the field of Music Information Retrieval (MIR) that focus on explainability for deep models consists mainly of feature-based post hoc methods (e.g., [3–5, 7]). Chowdhury et al. introduce pre-defined "mid-level features" that could be considered concepts as intermediate targets in a two-level prediction model [16]. Related to this, approaches that consider instrinsic as opposed to post-hoc methods gain increasing attention in the audio domain as well (e.g., [17–19]). However, no prior studies have examined post-hoc concept-based explainability techniques on systems that work with musical data. To provide a technical context for our work, we focus on related approaches that work on audio or image data.

A recent approach [20] applies concept-based techniques to multimodal data (video, audio, and text) to explain an emotion classifier for video sequences of human conversations. For the audio signal, they only test the concept of "voice pitch", i.e., the averaged fundamental frequency of the speaker's voice. In another related study, Parekh et al. [21] learn a codebook of sounds (e.g., alarm sound) from input audio through Non-negative Matrix Factorisation (NMF), which is then used to obtain hidden network layer representations that indicate time activations of these pre-learnt components. This has some similarities with our unsupervised approach, as we also make use of non-negative factorisation techniques to disentangle concepts. However, while [21] performs the factorisation on the input and propagates the results to a hidden layer, we factorise the hidden layer activations and project the results back to the input data. The approach of [21] is promising if we assume that the underlying reason for a system decision can be extracted directly from the input with unsupervised separation approaches. However, since this might not always be the case, we factorise the layer activations to exploit the non-linear feature extraction a network does internally to obtain more meaningful explanations.

Our work uses techniques and results originally proposed for the image domain. The work of Kim et al. [12] provides the basis for the supervised explanation, although the creation of concept data sets is more challenging for music. We base the unsupervised explanation on the work of Zhang et al. [14], but propose a dedicated visualisation of piece excerpts and test different solutions for the tensor factorisation step by employing the NTD.

[3] https://github.com/CPJKU/composer_concept

3. EXPERIMENTAL SETUP

This section details the type of data and the system that we use to demonstrate our explainability techniques.

Data: We use MIDI representations of piano performances from the MAESTRO v2.0.0 dataset [22]. As proposed by Kim et al. [15], we pre-select data by composers with at least 16 pieces and remove files with more than one composer (e.g., Schubert/Liszt, "Der Mueller und der Bach"), so that pieces of 13 different composers remain (see Table 1). We randomly split the resulting 667 pieces in a training (462 pieces) and validation (205 pieces) set. For each piece, we sample 90 excerpts of 20 seconds randomly across time and different performances of the same piece (if available).

Composer Classifier: In this work, we investigate the composer classification system proposed by Kim et al. [15]. For more recent systems, the code was not available [23] or we were unable to reproduce their results [24]. During preprocessing, we transform MIDI excerpts into piano roll representations with a 50 ms time step, i.e., a matrix 88×400, which is used as input to a ResNet-50 [25]. Kim et al. [15] use an additional channel with onset information, which we omit because it does not improve the performance of our system. As proposed by [15], we train the network with Stochastic Gradient Descent with momentum (factor 0.9), L2 weight regularisation (factor 0.0001), and cross-entropy loss function. The initial learning rate is set to 0.01, and scheduled with cosine annealing [26]. Our attempt to retrain the system results in a F1 score of 0.93 compared to 0.83 in the original work [15]. The accuracy of our system is 0.93. The difference in performance could be attributed to a problem during preprocessing in the original code, which reduced the resolution of the piano roll.

4. SUPERVISED CONCEPT-BASED EXPLANATIONS

In this section, we use TCAV [12] to build a supervised concept-based explainer. We manually define musical concepts and interrogate a music classifier to find out how much a concept influences the results of the classifier.

4.1 Musical Concepts

Musical concepts describe the characteristics of a certain group of notes and are identified by musicologists with a specific name or with a small sentence (e.g., "staccato", "rubato", "melody with jumps", etc.). To define a musical concept in a way that can be used within our system, we construct *concept datasets*, i.e., sets of pieces that have one specific musical concept in common. In this paper, we build three different concept datasets, each consisting of 30 musical excerpts of ~25 seconds. Ideally, the bigger and more diverse the concept datasets is, the lower is the probability that it will also represent other unwanted concepts.