

# TRANSFER LEARNING AND BIAS CORRECTION WITH PRE-TRAINED AUDIO EMBEDDINGS

Changhong Wang<sup>1</sup>      Gaël Richard<sup>1</sup>      Brian McFee<sup>2</sup>

<sup>1</sup>LTCI, Télécom Paris, Institut Polytechnique de Paris, France

<sup>2</sup>Music and Audio Research Laboratory, New York University, USA

changhong.wang@telecom-paris.fr

## ABSTRACT

Deep neural network models have become the dominant approach to a large variety of tasks within music information retrieval (MIR). These models generally require large amounts of (annotated) training data to achieve high accuracy. Because not all applications in MIR have sufficient quantities of training data, it is becoming increasingly common to transfer models across domains. This approach allows representations derived for one task to be applied to another, and can result in high accuracy with less stringent training data requirements for the downstream task. However, the properties of pre-trained audio embeddings are not fully understood. Specifically, and unlike traditionally engineered features, the representations extracted from pre-trained deep networks may embed and propagate biases from the model’s training regime.

This work investigates the phenomenon of bias propagation in the context of pre-trained audio representations for the task of instrument recognition. We first demonstrate that three different pre-trained representations (VGGish, OpenL3, and YAMNet) exhibit comparable performance when constrained to a single dataset, but differ in their ability to generalize across datasets (OpenMIC and IRMAS). We then investigate dataset identity and genre distribution as potential sources of bias. Finally, we propose and evaluate post-processing countermeasures to mitigate the effects of bias, and improve generalization across datasets.

## 1. INTRODUCTION

*Transfer learning* generally refers to the concept of adapting a model for one task to solve another task. Often, this is achieved by extracting the internal representation (an *embedding*) of input data from a pre-trained neural network, and providing it as input features to some (often simpler) *downstream* model for the target task. While this approach is increasingly common and effective, pre-trained embedding models may encode and propagate implicit biases which can have detrimental and disparate population-

dependent effects. Biases have caught wide attention from research fields such as natural language processing (NLP) [1–3], cognitive science [4], and computer vision [5], while in music information retrieval (MIR), bias of pre-trained audio embeddings, is under-explored.

In this paper, we identify and address the bias of different pre-trained audio embeddings for transfer learning on the task of instrument recognition. We summarize the contributions as following. (1) We study the within- and cross-domain performance of three pre-trained audio embeddings (VGGish, OpenL3, YAMNet) on two instrument datasets (IRMAS and OpenMIC-2018). (2) We demonstrate that this approach can propagate bias by producing classifiers which are sensitive to the source domain (dataset). (3) Based on the performance variation in cross-domain generalization, we investigate dataset identity and genre distribution as potential sources of bias. (4) We propose a post-processing countermeasure to mitigate unwanted bias in the representation. We experiment different bias correction strategies, and analyze the robustness of each pre-trained audio embedding. The proposed strategies make use of relatively little additional information, and generally produce a modest improvement to cross-domain accuracy for the instrument recognition task. Our code for all experiments is publicly available<sup>1</sup>.

## 2. RELATED WORK

Pre-trained embeddings are becoming increasingly used in transfer learning for audio-related tasks. Choi et al. [6] presented a transfer learning approach for music classification and regression tasks using the internal activations of a pre-trained convolutional network as features. The network was trained on the source task of music tagging, and the learned representation was then transferred to five target tasks, including genre classification, vocal/non-vocal classification, emotion prediction, speech/music classification, and acoustic event classification. Other well-known audio embedding models include OpenL3 [7], VGGish [8], and YAMNet<sup>2</sup>. The OpenL3 is a 512-dimensional embedding model that results from self-supervised training of the look-listen-learn (L3)-Net for audiovisual correlations. VGGish (128-dimensional) and YAMNet (1024-



© Changhong Wang, Gaël Richard, and Brian McFee. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** Changhong Wang, Gaël Richard, and Brian McFee, “Transfer Learning and Bias Correction with Pre-trained Audio Embeddings”, in *Proc. of the 24th Int. Society for Music Information Retrieval Conf.*, Milan, Italy, 2023.

<sup>1</sup> <https://github.com/changhongw/audio-embedding-bias>

<sup>2</sup> <https://github.com/tensorflow/models/tree/master/research/audioset/yamnet>

dimensional) are both embeddings derived from classification models trained on AudioSet [8]. Although these embedding models differ in the architecture of the network, source data, and training regime, they have each demonstrated good and comparable generalization performance for a variety of tasks.

Despite that embedding models are normally trained on large amounts of data, it inevitably encodes biases due to the limitation of collected data. This problem can be especially prevalent in models trained via unsupervised or self-supervised strategies, where there may be no incentive for the model to learn invariances or equivalencies in the data beyond what is required for by the training objective. As a result, pre-trained embeddings may propagate unwanted biases to downstream tasks. Different types of biases and bias correction methods are explored in the NLP literature, such as gender [9], race and religion [10]. A general approach for addressing gender bias in word embedding was proposed by Bolukbasi et al., following three steps: identify bias direction, remove bias by projecting out the bias direction, and equalize pairs [9].

Besides field-specific biases, dataset bias is a general type of bias that could happen in any application domain. Tommasi et al. [5] investigated dataset bias in visual recognition with a cross-dataset testbed comprising 12 different datasets. Ganin et al. [11] proposed adversarial training for domain adaptation to reduce sensitivity to data drawn from similar but different distributions. When detecting depression, a mental health disorder, from speech, Bailey and Plumbley [12] found that biases in dataset could result in skewed classification performance.

The approach we take in this paper is most similar to those of Bolukbasi et al. [9] and Ganin et al. [11]. While Bolukbasi et al.’s method requires numerous paired examples to identify a subspace which encodes undesirable bias, our proposed method works at the level of collection statistics rather than individual correspondence, and may be easier to apply for audio applications. Similarly, Ganin et al.’s method requires adversarial training of the initial model to produce a representation which cannot discriminate well between subsets of data that should be treated equivalently. Our approach is implemented as a post-processing step, and can be applied to any pre-trained model. While we focus in this work on dataset identity as a concrete source of bias, we emphasize that the method should be generally applicable to other scenarios in which audio representations exhibit unwanted sensitivity to identifiable attributes.

### 3. METHODS

We consider embedding bias from the perspective of *domain adaptation*. Unlike transfer learning, which relates to the *output* of the model, domain adaptation refers to the behavior of a model (classifier, regressor, etc.) under changes to the distribution of *input data*. This is closely related to *representation bias*, which is one among many forms of bias known to impact machine learning systems as enumerated by Mehrabi et al. [13]. If a classifier is trained on a sample of (labeled) data which is not representative of

the target population, then we expect the model to generalize poorly. The degree to which a pre-trained audio embedding is sensitive to differences between populations of interest—e.g., between a dataset annotated for instrumentation, compared to other collections of music—is therefore of principal interest [14].

#### 3.1 Domain sensitivity

We investigate the domain sensitivity of three pre-trained embeddings (OpenL3, VGGish, and YAMNet) in transfer learning for the downstream task of instrument recognition. Each embedding is evaluated in both within-domain and cross-domain setting. For within-domain evaluation, we train and test the embedding in a single dataset; while for cross-domain case, we investigate the domain adaptation capability of the embedding models across datasets, i.e. training and testing the downstream classifier using data from different datasets. As a study case, we consider two well-known datasets for instrument recognition, i.e. IRMAS [15] and OpenMIC-2018 [16] (see Section 4.1 for dataset details).

Fig. 1 (a) and (d) visualize the within-domain (IRMAS–IRMAS and OpenMIC–OpenMIC) recognition results in terms of area under the receiver operating characteristic curve (ROC-AUC) using the three embeddings above for each of the ten instrument classes.<sup>3</sup> All three embeddings achieve comparable results, although there is a loose performance ranking of YAMNet > OpenL3 > VGGish for most instrument classes.

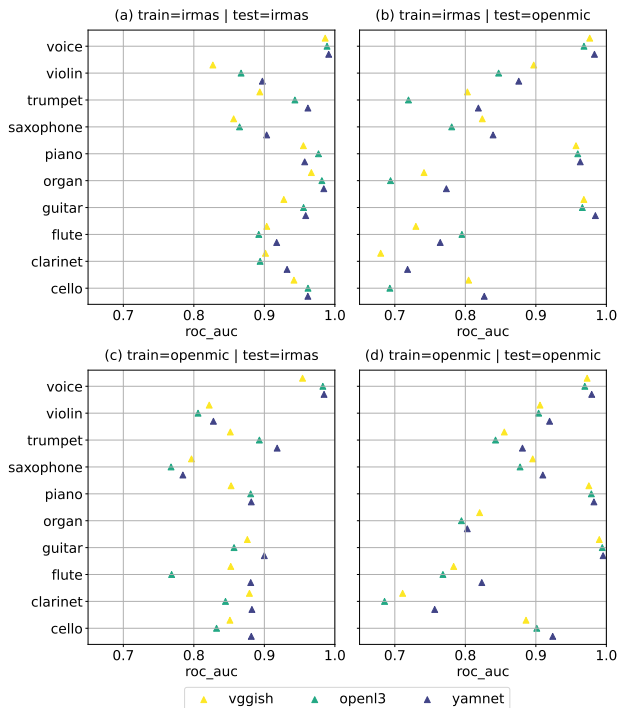
When generalizing across domains, performance degradation happens for both cross-dataset pairs, as shown in Fig. 1 (b) IRMAS–OpenMIC and (c) OpenMIC–IRMAS. The performance ranking of the three embedding does not persist either. Comparing the results when testing on OpenMIC, i.e. (b) and (d), only *voice*, *piano*, and *guitar* retain close results. For the remaining instrument classes, all three embeddings exhibit diminished performance. Similar trends take place in the comparison between (a) and (c) where the test set is IRMAS.

Surprisingly, a dramatic performance drop happens for the *organ* class. Examining this class in both datasets, we notice a large distribution difference on genre, as shown in Fig. 2. Organ in IRMAS is confined to *pop/rock* and *jazz/blue* genres, suggesting that examples mostly contain electric organ sounds (e.g. Hammond B3). The distribution of organ in OpenMIC is more balanced, but dominated by classical recordings which are more likely to contain pipe organ than electric. These differences aside, we generally expect the instrument labels to refer to similar sounds across domains.

#### 3.2 Quantifying domain bias

To quantify the effect of domain bias, we first obtain the domain separation direction vector  $w \in \mathbb{R}^D$  by fitting a linear discriminant analysis (LDA) model to discriminate

<sup>3</sup> We report AUC because it is invariant to overall class proportions and decision thresholds—which vary between datasets—and thereby allows us to focus on the separating directions identified for each class.



**Figure 1.** Within-domain (a, d) and cross-domain (b, c) performance of pre-trained audio embeddings (VGGish, OpenL3, and YAMNet) on instrument recognition in the IRMAS and OpenMIC datasets. ROC-AUC refers to area under the receiver operating characteristic curve.

between the OpenMIC and IRMAS datasets in each representation (VGGish, OpenL3, and YAMNet).  $v_k \in \mathbb{R}^D$  is the instrument separation direction vector, i.e. the coefficient vector of the trained downstream classifier, for the  $k$ -th instrument.  $k = 1, 2, \dots, K$  is the instrument class index and  $D$  is the dimension of pre-trained embedding. We measure the correlation between the domain separation and downstream classification using the cosine similarity between  $w$  and  $v_k$ :

$$c_k(w, v) = \frac{\langle w, v_k \rangle}{\|w\| \times \|v_k\|} \quad (1)$$

Large (in magnitude)  $c_k$  indicates that the instrument classifier is sensitive to dataset identity.

Fig. 3 top shows the absolute correlation value for each instrument class, when the classifier is trained on the training set of (a) IRMAS and (b) OpenMIC dataset, respectively. The mean correlation value over all instruments for each embedding is displayed in the legend. It clearly shows that YAMNet is the least sensitive to dataset bias; OpenL3 is also relatively stable while VGGish is the most sensitive to dataset bias. The relatively large correlation value for the organ class matches our analysis in Section 3.1 that genre distribution might be also a potential source of bias (see Fig. 2). Although the sensitivity of different embeddings to dataset bias are different, bias cannot generally be removed by simply using different pre-trained embeddings. As we will demonstrate, explicitly correcting for dataset bias can potentially improve domain adaptation

performance for each choice of embedding.

### 3.3 Bias correction

To mitigate domain bias, we propose a post-processing countermeasure on the pre-trained embeddings which does not interact with the training process of embeddings. Importantly, the proposed method requires only samples of data which should behave similarly for the downstream task, but it does not require these samples to be *labeled* for the downstream task.

Continuing our instrument classification example, given that both datasets contain examples from each of the instrument categories of interest, we should expect that a well-formed linear classifier should behave independently of the domain from which data is drawn. Concretely, this means that the separating direction  $v_k$  should be orthogonal to any direction  $w$  which separates the two datasets in the embedding space, resulting in  $c_k(w, v_k) = 0$ . While Ganin et al. [11] use this intuition to adversarially train the representation, this approach is impractical when using pre-trained embeddings which are presumed to be fixed in advance. Instead, we approach this problem by post-processing the embedding to project out the direction  $w$  which separates domains that should be indistinguishable for the downstream task.

Concretely, if  $w \in \mathbb{R}^D$  is the domain-separating direction (normalized to unit length,  $\|w\| = 1$ ), we project this dimension out of the space by applying the following transformation to input data  $x \in \mathbb{R}^D$ :

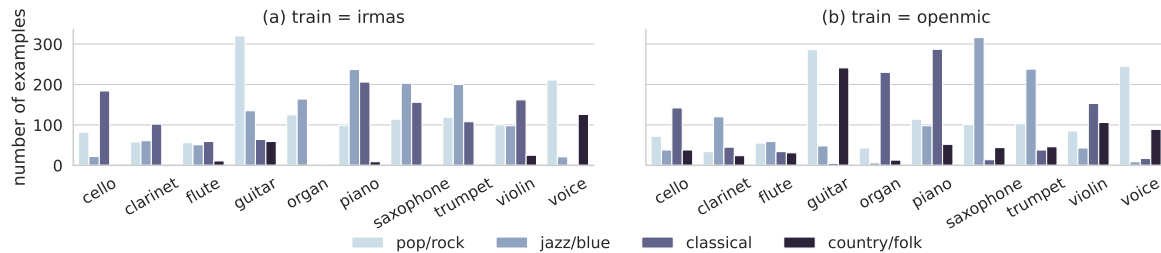
$$x_P := (\mathbf{I} - ww^T) x \quad (2)$$

where  $\mathbf{I}$  is the  $D \times D$  identity matrix. The new embedding  $x_P$  is the input to the classifier.

### 3.4 Multiple bias correction

While the above strategy is defined for binary bias correction, e.g. where there are two domains to be reconciled, it does generalize to more complex settings. In the instrument recognition example, we may also consider differences between genres across datasets as a source of bias. Even if two datasets both consist of examples in the same genre categories, this does not necessarily mean that the genre terms are used consistently between datasets.

To consider the influence of genre distribution, we propose also *multiple bias correction*, where we extract the dataset separation direction in the genre subspace. That is, for each pair of matched genre labels, e.g. pop/rock in IRMAS and pop/rock in OpenMIC, we fit a binary LDA to separate them. Then for each genre category  $g = 1, 2, \dots, G$  (for  $G \geq 1$  genres), we obtain a dataset separation direction vector  $w_g$  which depends only on examples from genre  $g$ . Collecting all  $w_g$  into a matrix  $W \in \mathbb{R}^{D \times G}$  defines a basis for a subspace of the embedding of dimension at most  $G$ . Note that  $W$  may not be an orthogonal basis, as different  $w_g$  may correlate with each other.



**Figure 2.** Number of genre examples for each instrument in the training set of IRMAS and OpenMIC datasets. We align the genre labels according to those in the IRMAS dataset: pop/rock, jazz/blue, classical, and country/folk.

We therefore derive an orthogonal basis by factorizing  $W$  via the reduced singular vector decomposition (SVD):

$$W = U\Sigma V^T \quad (3)$$

where  $\Sigma$  is a  $G \times G$  diagonal matrix of singular values, and  $U \in \mathbb{R}^{D \times G}$  and  $V \in \mathbb{R}^{D \times G}$  are the left- and right-singular vectors. We use the right singular vectors as an orthogonal basis for the domain-separating subspace, resulting in the following generalization of Eq. (2):

$$\mathbf{x}_P := (\mathbf{I} - VV^T) \mathbf{x} \quad (4)$$

In applying Eq. (4), it is important to verify that  $W$  is full rank ( $G$ ), e.g. by verifying that all singular values  $\Sigma$  are sufficiently large, as Eq. (4) would otherwise remove a larger than necessary subspace from the representation. In all cases studied in this work,  $W$  was full rank.

### 3.5 Nonlinear bias correction

The above methods are based on two assumptions: 1) that the downstream model will be linear, and 2) that the domains are linearly separable. These assumptions may be restrictive in practice, so we generalize the method above by transforming the embeddings to a higher dimensional space using kernel methods. While both logistic regression and linear discriminant analysis support kernel generalizations [17], the subspace projection method described above is less directly adaptable.<sup>4</sup>

Instead of using implicit kernel representations, we will use approximate, i.e. explicit kernel approximation. That is, instead of replacing inner products  $\langle \mathbf{w}, \mathbf{v} \rangle$  by nonlinear kernel function calculations  $k(\mathbf{w}, \mathbf{v})$ , we apply an explicit nonlinear transformation  $f : \mathbb{R}^D \rightarrow \mathbb{R}^{D'}$  such that

$$\langle f(\mathbf{w}), f(\mathbf{v}) \rangle \approx k(\mathbf{w}, \mathbf{v}) \quad (5)$$

We then apply the previously defined bias correction methods on the transformed data  $f(\mathbf{w})$ , which results in projecting out the dataset-separating direction(s) after applying  $f$  but prior to fitting the downstream (instrument) classifiers.

There are several choices to be made here when selecting the kernel  $k$  and the approximating map  $f$ . In this work, we use a standard radial basis function (Gaussian)

<sup>4</sup> One could achieve a similar effect by adding a linear constraint  $\langle \mathbf{w}, \mathbf{v} \rangle = 0$  to the logistic regression problem, but this would require a custom solver and limit the general utility of the approach.

kernel and the “random Fourier features” approximation method [18]. However, we note that other choices (e.g., the Nyström method) are readily available in scikit-learn [19], and may work just as well.

In total, we have four bias-correction strategies: linear bias correction (*LDA*), linear multiple bias correction (*mLDA*), nonlinear bias correction in the kernelized embedding space (*KLDA*), nonlinear multiple bias correction in the kernelized embedding space (*mKLDA*).

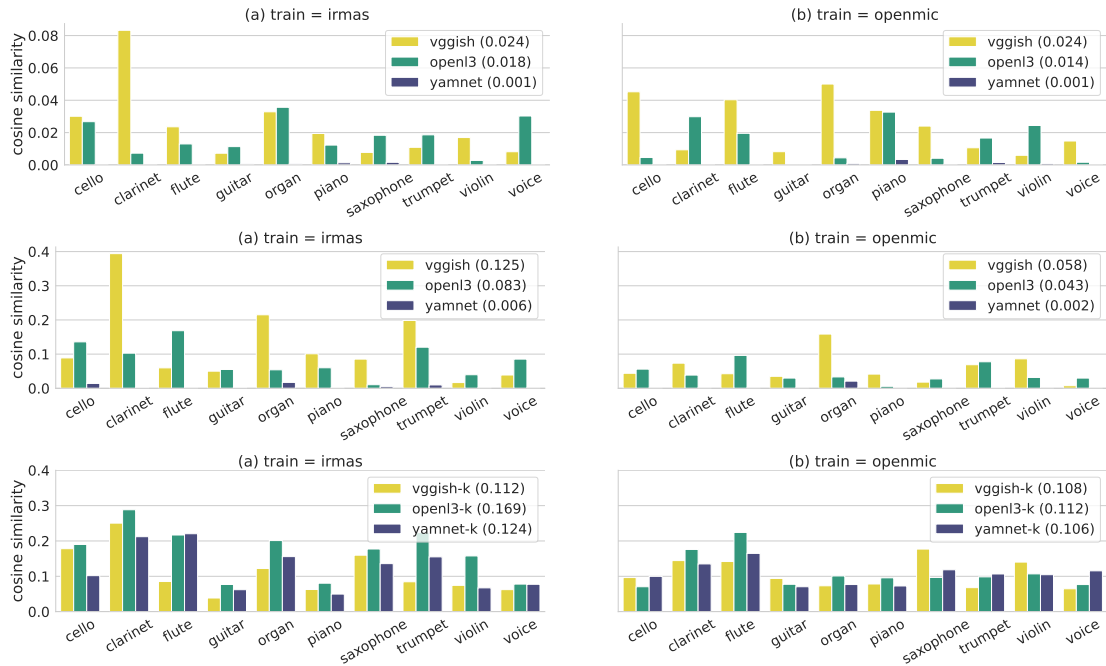
## 4. EXPERIMENTS

### 4.1 Datasets and experimental details

The datasets we use are two well-known datasets with instrument annotations, IRMAS [15] and OpenMIC-2018 [16]. The former comprises 20,000 examples of 10-second excerpts, partially labeled for the presence or absence of 20 instrument classes; and the latter contains 6705 audio files of 3-second clips, of which only the predominant instrument were annotated. Since there are 20 instrument classes in the OpenMIC dataset and 11 in the IRMAS, we focus only on the 10 mutual classes: *cello*, *clarinet*, *flute*, *guitar*, *organ*, *piano*, *saxophone*, *trumpet*, *violin*, and *voice*. For the sake of consistency, *electric guitar* and *acoustic guitar* in the IRMAS dataset have been merged into a single class: *guitar*.

To investigate the impact of genre, we also align the genres in the two datasets. Each audio sample in the IRMAS dataset is labeled with one of the five genres: *pop/rock*, *jazz/blue*, *classical*, *country/folk*, and *latin/soul*; while samples in the OpenMIC dataset has multiple labels from around 130 genres. We consider four genres (pop/rock, jazz/blue, classical, country/folk) as the latin/soul genre has few examples in both datasets. The genre labels of the OpenMIC dataset are merged into those of IRMAS with name intersections. For example, we merge the genres—*Rock*, *Loud-Rock*, *Noise-Rock*, *Psych-Rock*, et. al.—in OpenMIC into one genre label: *pop/rock*. Multiple genre labels in the OpenMIC dataset are reduced to a single label by the first activation from the four considered genres or the first of the original labels otherwise.

With the embedding features extracted using pre-trained VGGish, OpenL3, and YAMNet models, we train a logistic regression classifier for each instrument class using IRMAS and OpenMIC training data. The input to the



**Figure 3.** Correlation between domain separation and instrument classification for each instrument in the IRMAS and OpenMIC training set. (*Top*): correlation in the original embedding space with domain separation direction extracted using only dataset identity; (*middle*): correlation in the original embedding space with domain separation direction extracted class-wise; (*bottom*): same as middle but in the kernelized embedding space. Mean value is given in parentheses.

classifier is the mean frame embedding of each audio example. For OpenMIC dataset, we follow the train-test split in [16], with a ratio of 3:1. For fair comparisons, we create a new partition with the same train-test ratio on IRMAS dataset which takes into account of the class-balance and non-track overlap between training and test sets. To focus only on distribution shift, we use the same number of samples per class for both datasets during training, following the lower one.

For the nonlinear method, we first standardize the embedding features using z-score normalization with the training-set statistics. Then we approximate the kernels for the embeddings with a fixed dimension  $D'$  of four times the dimension ( $D$ ) of the original embeddings. Finally, we tune the hyper-parameter for the logistic regression classifier, i.e. the inverse of regularization strength  $C$ , by cross-validation with a grid of  $10^{-8:1:4}$ .

## 4.2 Results

Table. 1 lists the instrument classification performance of the debiasing methods discussed in Section 3 in terms of mean ROC-AUC over the 10 instrument classes. To compare the performance of using only dataset identity as additional information and that uses also class-labels, we present two sets of results: *global bias correction* and *class-wise bias correction*. We first present some observations that are common to both cases and then discuss the comparison. For the original embeddings (in *italic*), large performance drop shows for all cross-domain cases. OpenL3 is most sensitive to distribution shift, with a drop of 12.7% and 7% when testing on IRMAS and OpenMIC

dataset, respectively. Yet, from the cosine similarity values in Fig. 3 top and middle, OpenL3 does not embed the most domain bias. This may indicate that for the task at hand, other more significant distribution shifts that OpenL3 is sensitive to may exist. For all embeddings, projecting to the higher dimensional space (debiasing methods with “K”) almost never substantially hurts the within-domain performance and sometimes improves the performance.

Interestingly, when comparing linear debiasing (“-LDA” and “-mLDA”) with nonlinear debiasing (“-KLDA” and “-mKLDA”) for all embeddings, we find that kernelization does not help for VGGish while YAMNet only works in the kernelized embedding space. This explains the relative increase of cosine similarity values for YAMNet after kernelization as compared to the other two embeddings (see Fig. 3 bottom). Both linear and nonlinear debiasing exhibit performance improvement for OpenL3. In terms of global bias correction, almost no improvement for VGGish except LDA for OpenMIC->IRMAS; OpenL3 yields some boost for both cross-domain cases. YAMNet improves the results only for OpenMIC->IRMAS. It is expected that the class-wise bias correction achieves better performance as we extract the domain bias for the target instrument exactly. This is also verified by the more noticeable cosine similarity values in the middle subfigure as compared to the top of Fig. 3. VGGish and OpenL3 yields slight improvement for most linear debiasing. All nonlinear debiasing improves the results of OpenL3 for IRMAS->OpenMIC and YAMNet for OpenMIC->IRMAS. Although the overall improvement is not significant, we observe large improvements for some instrument classes.

Debiasing method	Global bias correction				Class-wise bias correction			
	Within-domain		Cross-domain		Within-domain		Cross-domain	
	IR-IR	OP-OP	OP-IR	IR-OP	IR-IR	OP-OP	OP-IR	IR-OP
<i>VGGish</i>	<i>91.6</i>	<i>87.95</i>	<i>82.82</i>	<i>83.81</i>	<i>91.60</i>	<i>87.95</i>	<i>82.82</i>	<i>83.81</i>
VGGish-LDA	91.60	87.99	82.99 ( <b>+0.18</b> )	83.82 (0.0)	91.60	87.94	82.93 ( <b>+0.12</b> )	83.85 (+0.03)
VGGish-mLDA	91.45	87.98	82.70 (-0.11)	83.30 (-0.51)	91.56	87.87	83.13 ( <b>+0.31</b> )	83.66 (-0.16)
VGGish-K	92.24	88.08	82.57 (-0.25)	83.67 (-0.14)	92.24	88.08	82.57 (-0.25)	83.67 (-0.14)
VGGish-KLDA	92.24	88.08	82.58 (-0.24)	83.67 (-0.14)	92.22	88.07	82.70 (-0.12)	83.78 (-0.04)
VGGish-mKLDA	92.22	88.15	82.42 (-0.39)	83.70 (-0.11)	92.26	88.08	82.70 (-0.11)	83.76 (-0.05)
<i>OpenL3</i>	<i>93.26</i>	<i>87.16</i>	<i>80.56</i>	<i>80.13</i>	<i>93.26</i>	<i>87.16</i>	<i>80.56</i>	<i>80.13</i>
OpenL3-LDA	93.26	87.16	80.56 (+0.01)	80.15 (+0.02)	93.24	87.18	80.59 (+0.04)	80.38 ( <b>+0.26</b> )
OpenL3-mLDA	93.11	87.16	80.67 ( <b>+0.12</b> )	79.93 (-0.20)	93.09	87.23	80.57 (+0.02)	80.62 ( <b>+0.50</b> )
OpenL3-K	93.89	87.91	79.46 (-1.09)	81.23 ( <b>+1.11</b> )	93.89	87.91	79.46 (-1.09)	81.23 ( <b>+1.11</b> )
OpenL3-KLDA	93.89	87.84	79.03 (-1.53)	81.23 ( <b>+1.11</b> )	93.96	87.91	79.99 (-0.57)	81.79 ( <b>+1.66</b> )
OpenL3-mKLDA	93.88	87.88	79.56 (-1.00)	81.20 ( <b>+1.07</b> )	94.04	87.83	79.97 (-0.59)	81.32 ( <b>+1.19</b> )
<i>YAMNet</i>	<i>94.65</i>	<i>89.74</i>	<i>85.01</i>	<i>85.47</i>	<i>94.65</i>	<i>89.74</i>	<i>85.01</i>	<i>85.47</i>
YAMNet-LDA	94.65	89.74	85.01 (0.0)	85.47 (0.0)	94.65	89.74	85.02 (0.0)	85.47 (0.0)
YAMNet-mLDA	94.65	89.74	85.01 (0.0)	85.47 (0.0)	94.65	89.74	85.02 (0.0)	85.46 (0.0)
YAMNet-K	93.83	89.24	85.87 ( <b>+0.86</b> )	84.56 (-0.91)	93.83	89.24	85.87 ( <b>+0.86</b> )	84.56 (-0.91)
YAMNet-KLDA	93.83	89.23	85.87 ( <b>+0.86</b> )	84.56 (-0.91)	93.63	89.24	86.00 ( <b>+0.99</b> )	84.76 (-0.70)
YAMNet-mKLDA	93.79	89.19	85.72 ( <b>+0.71</b> )	84.43 (-1.04)	93.79	89.34	85.53 ( <b>+0.51</b> )	84.60 (-0.87)

**Table 1.** Mean ROC-AUC (%) of global bias correction and class-wise bias correction on instrument classification in IRMAS (IR) and OpenMIC (OP) datasets. VGGish, OpenL3, and YAMNet (in *italic*) refers to the original embedding; the other cases, i.e. with -LDA, -mLDA, -LDA, and -mKLDA, correspond to linear, linear-multiple, nonlinear, and nonlinear-multiple debiasing strategies; cases with -K are the kernelized embeddings. Values in parenthesis are the performance boost (>0.1 are **bolded**) or degradation as compared to the original embedding (the closest underlined above).

## 5. DISCUSSION

We notice two important factors for transfer learning with pre-trained audio embeddings: the training regime of the embeddings, and the class vocabulary alignment between the source task and downstream task.

The better generalization performance of YAMNet and VGGish in a transfer setting may be attributed to their training regime. YAMNet and VGGish are derived from supervised training while OpenL3 is from self-supervised training and more prone to overfitting a domain. As a result, YAMNet and VGGish have both been incentivized to learn invariances within specific categories (including musical instrumentation), while OpenL3 has no such incentive as it is only designed to predict audio-visual correspondence. Moreover, YAMNet was specifically trained for sound classification using a vocabulary that broadly subsumes that of our downstream task (instrumentation). This likely contributes to its high performance and cross-domain stability overall.

The class vocabulary alignment is related to *label shift*, an under-explored type of distribution shift in the domain-adaptation field [20]. The labelling scheme difference between the two datasets complicates the debiasing as the IRMAS dataset only contains labels for the predominant instrument while all active instruments are annotated in the OpenMIC dataset. Aligning these two datasets is nontrivial as it involves label shift besides *covariate shift*. We propose multiple-bias correction, i.e. debiasing in the genre subspace, to deal with this problem. Yet, it does not resolve the conditional probability shift that happens due to unbalanced relationships between instrumentation and genre, e.g. the strong dependence between organ and pop/jazz

in IRMAS, and in this specific case an argument could be made that the classification task is closer to transfer learning than domain adaptation.

A notable limitation of the presented experiments is the small amount of functional data. Although OpenMIC dataset is relatively large with 14915 samples for training, only a small portion is actually used in the binary classification of each instrument. After equalizing the number of samples per class in both datasets, there are only 288, 221, 177, 578, 290, 551, 476, 427, 385, and 358 samples for the 10 instrument classes in the binary classification. Most classes have number of samples less than the dimension of OpenL3 (512) and all of them are below that of YAMNet (1024).

## 6. CONCLUSION

The method proposed in this work addresses one specific form of bias that can arise in transfer learning scenarios. Correctly applying this method requires identifying subsets of data that should be treated equivalently, i.e., be indistinguishable under the chosen representation. We stress that this notion of equivalence ultimately depends on the choice of the downstream task, and caution should be exercised when identifying populations to treat as equivalent. For the case study presented here—domain adaptation and instrument recognition—we argue that the downstream task ought to be generally independent of the source domain, though we recognize that this will not always be true in practice. We therefore urge practitioners to critically investigate all assumptions of equivalence and independence when applying bias correction methods.

## 7. ACKNOWLEDGMENTS

This work was partly funded by the European Union (ERC, HI-Audio, 101052978). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them. We thank Michel Olvera for the discussions on domain adaptation.

## 8. REFERENCES

- [1] I. Garrido-Muñoz, A. Montejo-Ráez, F. Martínez-Santiago, and L. A. Ureña-López, “A survey on bias in deep NLP,” *Applied Sciences*, vol. 11, no. 7, p. 3184, 2021.
- [2] E. Ntoutsis, P. Fafalios, U. Gadiraju, V. Iosifidis, W. Nejdl, M.-E. Vidal, S. Ruggieri, F. Turini, S. Papadopoulos, E. Krasanakis *et al.*, “Bias in data-driven artificial intelligence systems—an introductory survey,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 3, p. e1356, 2020.
- [3] N. Garg, L. Schiebinger, D. Jurafsky, and J. Zou, “Word embeddings quantify 100 years of gender and ethnic stereotypes,” *Proceedings of the National Academy of Sciences*, vol. 115, no. 16, pp. E3635–E3644, 2018.
- [4] M. G. Haselton, D. Nettle, and D. R. Murray, “The evolution of cognitive bias,” *The handbook of evolutionary psychology*, pp. 1–20, 2015.
- [5] T. Tommasi, N. Patricia, B. Caputo, and T. Tuytelaars, “A deeper look at dataset bias,” in *Domain adaptation in computer vision applications*. Springer, 2017, pp. 37–55.
- [6] K. Choi, G. Fazekas, M. B. Sandler, and K. Cho, “Transfer learning for music classification and regression tasks,” in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2017, pp. 141–149.
- [7] A. Cramer, H. Wu, J. Salamon, and J. 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.
- [8] A. Jansen, J. F. Gemmeke, D. P. Ellis, X. Liu, W. Lawrence, and D. Freedman, “Large-scale audio event discovery in one million YouTube videos,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 786–790.
- [9] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, “Man is to computer programmer as woman is to homemaker? debiasing word embeddings,” *Advances in neural information processing systems*, vol. 29, 2016.
- [10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, “Language models are few-shot learners,” *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- [11] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, “Domain-adversarial training of neural networks,” *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.
- [12] A. Bailey and M. D. Plumbley, “Gender bias in depression detection using audio features,” in *Proceedings of the IEEE European Signal Processing Conference (EUSIPCO)*, 2021, pp. 596–600.
- [13] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, “A survey on bias and fairness in machine learning,” *ACM Computing Surveys (CSUR)*, vol. 54, no. 6, pp. 1–35, 2021.
- [14] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, “Analysis of representations for domain adaptation,” *Advances in neural information processing systems*, vol. 19, 2006.
- [15] J. J. Bosch, J. Janer, F. Fuhrmann, and P. Herrera, “A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals,” in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp. 559–564.
- [16] E. Humphrey, S. Durand, and B. McFee, “Openmic-2018: An open data-set for multiple instrument recognition,” in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2018, pp. 438–444.
- [17] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer, 2009.
- [18] A. Rahimi and B. Recht, “Random features for large-scale kernel machines,” *Advances in neural information processing systems*, vol. 20, 2007.
- [19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [20] K. Azizzadenesheli, A. Liu, F. Yang, and A. Anandkumar, “Regularized learning for domain adaptation under label shifts,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.