



# How to Construct Perfect and Worse-than-Coin-Flip Spoofing Countermeasures: A Word of Warning on Shortcut Learning

Hye-jin Shim<sup>1</sup>, Rosa González Hautamäki<sup>2</sup>, Md Sahidullah<sup>3</sup>, Tomi Kinnunen<sup>1</sup>

<sup>1</sup>University of Eastern Finland, Finland

<sup>2</sup>University of Oulu, Finland

<sup>3</sup>TCG CREST, India

hyejin.shim@uef.fi, rosahautamaki@gmail.com, sahidullahmd@gmail.com,  
tomi.kinnunen@uef.fi

## Abstract

Shortcut learning, or ‘Clever Hans effect’ refers to situations where a learning agent (e.g., deep neural networks) learns spurious correlations present in data, resulting in biased models. We focus on finding shortcuts in deep learning based spoofing countermeasures (CMs) that predict whether a given utterance is spoofed or not. While prior work has addressed specific data artifacts, such as silence, no general normative framework has been explored for analyzing shortcut learning in CMs. In this study, we propose a generic approach to identifying shortcuts by introducing systematic interventions on the training and test sides, including the boundary cases of ‘near-perfect’ and ‘worse than coin flip’ (label flip). By using three different models, ranging from classic to state-of-the-art, we demonstrate the presence of shortcut learning in five simulated conditions. We also analyze the results using a regression model to understand how biases affect the class-conditional score statistics.

**Index Terms:** dataset bias, shortcut learning, Clever Hans, anti-spoofing, ASVspooF

## 1. Introduction

The study of deep learning models has increased along with their widespread adoption in applications processing large amounts of data [1–5]. However, unexpected model behavior or system outcomes can be incurred when simply enlarging the scale of datasets and achieving high numerical accuracy without thorough examinations of the data and models. Accordingly, warnings have been raised about the potential risks associated with *biased* datasets and models [6–8]. This kind of biased model behavior based on spurious correlations is referred to as *Clever Hans effect* [9] or *shortcut learning* [10].

Several studies have been conducted to understand how models work and uncovered potential biases in the data [11–17]. For example, the models in the image domain are prone to focus on the background, rather than the object as a shortcut to predict class labels [18–20]. The situation becomes more complex when using a deep-learning *black-box* model, which is difficult to interpret. *Explainable AI* (XAI) [21] is one of the ways to interpret model behavior and build human-understandable models [22–24].

In this study, we propose a novel framework to discover shortcut learning in binary classifiers, treated as black-boxes. The proposed framework, detailed in Section 2, formulates a generic approach for introducing systematic biases into an existing database. Purposefully introduced interventions in *asymmetric* ways lead the model to respond to the provided shortcuts. Those interventions are applied both in the training and test sides, including the boundary cases of ‘near-perfect’ and ‘worse than coin flip’ cases (label flip). Since the parameters

Table 1: We consider biased audio anti-spoofing task with two classes (spoof and bona fide) where specific interventions are applied to one of the four data subsets as indicated. Besides the original dataset protocol (**O**) without interventions, we consider four biased configurations labeled **A** through **D**. In **A** and **B**, the treatment for the same class across training and test subsets are identical; in **C** and **D**, the treatments are flipped. Training and evaluating models on biased data allows addressing their sensitivity to unintended ‘shortcuts’. We expect a model to yield lower error rates on **A** and **B** compared to **O**; and higher error rate on **C** and **D** compared to **O**.

Configuration (indicator)	Train		Test	
	Spf	Bona	Spf	Bona
<b>O</b> (0 0 0 0)				
<b>A</b> (0 1 0 1)		✓		✓
<b>B</b> (1 0 1 0)	✓		✓	
<b>C</b> (0 1 1 0)		✓	✓	
<b>D</b> (1 0 0 1)	✓			✓

of the intervention process are known, another key element of the proposed approach is to use these parameters as inputs in a regression model (linear mixed effects modeling or LME) in the classifier score domain. Importantly, the inclusion of LME allows us to ‘go beyond the EER’ – to learn how the biases impact the class-conditional score statistics.

The focus of our case study is on audio anti-spoofing, which determines whether the utterance is from a real human (bona fide) or spoofing attacks (e.g. voice conversion, text-to-speech). Several studies have addressed data bias in audio anti-spoofing [25–30]. The early studies in [25–27] have investigated the distribution of waveform samples as a shortcut in spoofing countermeasures (CM). More recently, the validity of the ASVspooF 2019 and ASVspooF 2021 datasets has also raised concern about the proportion of silence in [28, 31]. While *silence* has only been examined in spoofing detection yet, our research explores several suspected candidates for leading bias.

Our proposed shortcut learning analysis framework is applicable to arbitrary types of data interventions and black-box classifiers. As a proof of concept, therefore, our experimental part includes five different types of interventions and three different spoofing countermeasures (CMs) of varied complexity. In particular, experiments are conducted with two conventional methods, namely, Gaussian mixture model (GMM) and light convolutional network (LCNN) as well as state-of-the-art audio anti-spoofing using integrated spectro-temporal graph attention network (AASIST).

## 2. Methodology

To simulate and analyze data bias in a delicate way, we address binary classification (detection) where the system under study potentially learns unintended associations between class labels and extrinsic interventions. We consider carefully selected sets of systematic interventions designed to create asymmetry between the two classes, as presented in Table 1. This section details the proposed methodology.

### 2.1. Constructing and Evaluating Binary Classifiers

Let  $\mathcal{D} \triangleq \{(x_i, y_i^{\text{cls}}) : i = 1, \dots, N\}$  denote a labeled dataset of  $N$  objects  $x_i \in \mathcal{X}$  and their ground-truth class labels  $y_i^{\text{cls}} \in \mathcal{Y} \triangleq \{0, 1\}$ . The  $N$  instances are considered as independent, identically-distributed draws from an unknown data distribution  $P(X, Y)$ . In this study, each  $x_i$  is a speech utterance (a digital waveform) and  $y_i^{\text{cls}}$  indicates whether  $x_i$  is a bona fide ( $y_i^{\text{cls}} = 1$ ) or a spoofed ( $y_i^{\text{cls}} = 0$ ) waveform. The dataset  $\mathcal{D}$  consists of disjoint training and evaluation subsets denoted by  $\mathcal{D}_{\text{tm}}$  and  $\mathcal{D}_{\text{eva}}$ , respectively<sup>1</sup>. We introduce another binary variable  $y_i^{\text{tm}} \in \mathcal{Y}$  that indicates whether  $x_i$  belongs to the training ( $y_i^{\text{tm}} = 1$ ) or the evaluation ( $y_i^{\text{tm}} = 0$ ) subset. The two labels,  $\mathbf{y}_i \triangleq (y_i^{\text{cls}}, y_i^{\text{tm}}) \in \{0, 1\} \times \{0, 1\}$  specify both the class label (bona fide or spoof) and the use case (training or evaluation) of  $x_i$  in the dataset. These labels remain fixed (as-given) in the original dataset.

By using a selected class of predictive models  $g : \mathcal{X} \rightarrow \mathcal{Y}$  and training loss  $\mathcal{L} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}_+$ , a system developer trains a model by minimizing  $\mathcal{L}$  on the training data. The trained model is then executed on the evaluation set to make predictions of class labels. In practice, the model produces *score*  $s_i \in \mathbb{R}$  for each instance  $x_i \in \mathcal{D}_{\text{eva}}$ . The score can then be converted to a predicted hard decision by comparing  $s_i$  to a threshold  $\tau$ . The scores paired up with their ground-truth labels ( $s_i, y_i^{\text{cls}}$ ) are then used to compute miss and false alarm rates, each being a function of  $\tau$ . From the miss and false alarm rate functions, the evaluator summarizes performance using a suitable figure of merit – here, the equal error rate (EER). Besides EER, which measures discrimination, we also do direct modeling of the detection scores through linear mixed effects modeling as will be detailed below in Section 2.4.

### 2.2. Introducing Controlled (but Random) Interventions

While the ground-truth labels  $\mathbf{y}_i$  will remain as-given in the original data  $\mathcal{D}$ , we introduce random (but controlled) interventions to the audio files  $x_i$ . Our aim is to understand their impact on learning, predictions, and performance evaluation of models.

Consider an arbitrary file  $x_i$ . It gets randomly perturbed as

$$x'_i = f_j(x_i; z_{ij}), \quad (1)$$

with probability  $P(f_j) \triangleq \mathbb{P}(\text{Apply } f_j \text{ to } x_i)$ . Here  $f_j$  is a pre-defined, deterministic function of its inputs that takes different forms depending on the type of intervention (detailed in Subsection 3.2), indexed by the subscript  $j$ . The conditioning variable  $z_{ij} \sim_{\text{i.i.d.}} P(Z; \theta_j)$ , in turn, contains parameters of a random intervention applied to  $x_i$ . Using additive white noise as an example,  $z_{ij}$  might indicate the signal-to-noise ratio (SNR) selected for degrading waveform  $x_i$ .

<sup>1</sup>Additionally, when training models, we have a *development set*. Since in our modeling this is always treated the same way as the training set, we do not explicitly write notations about development data.

To sum up, the intervention process is specified by two sets of control parameters: (1) the probability of applying intervention,  $P(f_j)$ ; and (2) the distribution of the intervention control variable,  $P(Z; \theta_j)$ . The former controls *how many of the files in a given dataset get perturbed*. For a dataset of  $N$  files, we apply  $f_j$  to randomly selected  $N_{\text{perturb}} = \lfloor P(f_j) \cdot N \rfloor$  files, the remainder  $N - N_{\text{perturb}}$  being retained as in the original dataset. For the randomly selected files that undergo intervention, we sample another variable  $z_{ij} \sim P(Z; \theta_j)$  independent of  $x_i$  which controls the intervention applied to  $x_i$ .

### 2.3. Choosing Intervention Hyperparameters

For reasons of space and avoidance of combinatorial explosion, in this initial study, we make a number of simplifications about the interventions. First, while the selected types of interventions are typical in the audio domain, we focus on each type at a time; we neither mix different interventions into an experiment nor consider combinations of different interventions applied sequentially. Second,  $P(Z; \theta_j)$  takes the form of either a continuous uniform or a Dirac distribution. In the former case,  $P(Z; \theta_j)$  is specified by the minimum and maximum values, whereas the latter implies a deterministic choice of  $z_{ij}$ . An example of the former is white noise with randomly-selected SNR, and an example of the latter is  $\mu$ -law encoding.

For the intervention probability we consider the two extremes,  $P(f_j) \in \{0, 1\}$ , i.e. we either perturb *all* files ( $P(f_j) = 1$ ) or *none* ( $P(f_j) = 0$ ). Importantly, however, the intervention probability varies depending on the specific subset of data, as shown in Table 1. Introducing these *systematic effects* at the level of data subsets allows us to analyze a classifier’s behavior under the biased data set-up. From here on, we refer to the different 4-bit strings shown in the rows of Table 1 as *intervention configurations*, denoted by  $\mathbf{c} = (c_1, c_2, c_3, c_4) \in \{0, 1\}^4$ . Note that while the configuration remains fixed for a given training-evaluation experiment, the audio interventions are random at the level of *individual audio files* (except for  $\mu$ -law encoding).

### 2.4. Mixed Effects Modeling of Biased CM Scores

Suppose that a model  $g$  has been trained and scored on a dataset  $\mathcal{D}$  corresponding to a number of different configurations.

Since we have full knowledge of the intervention parameters, our aim is to define an *explanatory model* that links the detection scores to these parameters. Although observing changes in EER with different interventions being applied provides a quick overall trend, it does not provide details of how the cores are impacted per class. Hence, direct regression modeling of the CM detection scores as a function of intervention parameters helps to ‘go beyond the EER’ and obtain explanations for the impact of interventions.

To this end, our selected methodology consists of *linear mixed effects* (LME) modeling [32] of CM scores. We fit the following model separately for each CM and intervention type:

$$s_i = \underbrace{\mu + d y_i^{\text{cls}} + \beta^{\text{bona}} \Delta_i^{\text{bona}} + \beta^{\text{spf}} \Delta_i^{\text{spf}}}_{\text{fixed effect}} + \underbrace{\varepsilon_i}_{\text{random effect}} \quad (2)$$

Here,  $\mu$  is the global mean of the scores,  $d$  is a class discrimination parameter,  $y_i^{\text{cls}}$  the class label,  $(\beta_j^{\text{bona}}, \beta_j^{\text{spf}})$  two regression coefficients, and  $\varepsilon_i \sim \mathcal{N}(0, \sigma_\varepsilon^2)$  is a random effect that models variation across the trials. The model parameters, obtained by fitting (2) to CM score data, are  $(\mu, d, \beta^{\text{bona}}, \beta^{\text{spf}}, \sigma_\varepsilon^2)$ .

Table 2: Models per configuration and trial class  $y_i^{\text{cls}}$  (0: spoof, 1: bonafide), where Difference refers to the difference of conditional means:  $E[s_i|y_i = 1] - E[s_i|y_i = 0]$

Config.	$y_i^{\text{cls}}$	Model	Difference
<b>O</b>	0	$s_i = \mu + \varepsilon_i$	$d$
	1	$s_i = \mu + d + \varepsilon_i$	
<b>A, B</b>	0	$s_i = \mu + \beta^{\text{bona}} + \varepsilon_i$	$d + \beta^{\text{spf}} - \beta^{\text{bona}}$
	1	$s_i = \mu + d + \beta^{\text{spf}} + \varepsilon_i$	
<b>C, D</b>	0	$s_i = \mu + \beta^{\text{spf}} + \varepsilon_i$	$d + \beta^{\text{bona}} - \beta^{\text{spf}}$
	1	$s_i = \mu + d + \beta^{\text{bona}} + \varepsilon_i$	

Most relevant to the analysis of biased scores are the two variables  $\Delta_i^{\text{bona}}$  and  $\Delta_i^{\text{spf}}$ . The first one,  $\Delta_i^{\text{bona}}$ , is the absolute difference between the intervention probability of the test trial and the intervention probability of the bona fide training set. Likewise,  $\Delta_i^{\text{spf}}$  is the absolute difference between the intervention probability of the test trial and the intervention probability of the spoof training set. The value 0 indicates an equivalent treatment of test and training audio, while 1 indicates different treatments.

A concrete example may be helpful in clarifying (2). For configuration **A** in Table 1,  $\Delta_i^{\text{bona}} = 0$  and  $\Delta_i^{\text{spf}} = 1$  for all bona fide trials; and  $\Delta_i^{\text{bona}} = 1$  and  $\Delta_i^{\text{spf}} = 0$  for all spoof trials. The two class-conditional models obtained from (2) are

$$\begin{aligned} s_i &= \mu + \beta^{\text{bona}} + \varepsilon_i && (\text{spoof trials, } y_i^{\text{cls}} = 0) \\ s_i &= \mu + d + \beta^{\text{spf}} + \varepsilon_i && (\text{bona fide trials, } y_i^{\text{cls}} = 1) \end{aligned} \quad (3)$$

Since  $\varepsilon_i$  is normal, both of these conditional score distributions are normal as well, with shared variance  $\sigma_\varepsilon^2$ . The difference between the bona fide and spoof class means (which relates to discrimination) is  $d + (\beta^{\text{spf}} - \beta^{\text{bona}})$ . The expression in the parentheses vanishes when the two classes are treated the same way (original configuration **O**). Whenever the difference of  $\beta^{\text{spf}}$  and  $\beta^{\text{bona}}$  is positive, the separation of the two distributions improves, leading to ‘decreased’ EER relative to **O**. Likewise, a negative difference yields an ‘increase’ in EER. The use of quotes is intentional, as the  $\beta$ -coefficients relate to the systematic biases that we introduced to the data.

Similar model interpretations are easy to obtain for the remaining configurations; see the summary in Table 2. As one might have expected, the model for configurations **A** and **B** are the same; likewise, the model for the two label-flip configurations **C** and **D** are the same. Considering the difference of the class-conditional means, the only difference between the two sets of biased models is in the sign of  $(\beta^{\text{spf}} - \beta^{\text{bona}})$  or  $(\beta^{\text{bona}} - \beta^{\text{spf}})$ . In our result analysis, we use  $\beta^*$ , referring to  $\beta^{\text{bona}}$  or  $\beta^{\text{spf}}$ , as they only differ in sign.

## 3. Experimental setup

### 3.1. Dataset

We use the ASVspoof 2019 logical access (LA) dataset [5] for all the experiments. It consists of speech synthesis and voice conversion samples distributed across training, development, and evaluation subsets. The two former include six types of spoofing attacks, while the last contains thirteen attacks.

### 3.2. Selected interventions

We consider five different types of dataset interventions. They are motivated either by general considerations of the

desirable robustness of countermeasures; or reported findings on ASVspoof dataset biases.

**Mp3 compression** lowers the quality of the audio by MP3 encoder. We process audio files with *lameenc* encoder and bit-rates were randomly selected with varying quality ranging from 16 kbps to 256 kbps. Then, speech files are decoded again.

**Additive white noise** degrades speech files of the corpus with a random signal-to-noise-ratio value chosen between [0, 30] dB.

**Loudness normalization** utilizes a constant gain to match a specific loudness in *loudness units relative to full scale* (LUFS) which is an implementation of ITU-R BS.1770-4 [33]. The minimum and maximum values of loudness are -31 and -13 in dB and it is randomly selected for each sample in our work.

**Non-speech zeroing** sets a desired proportion of detected non-speech frames to a constant value (blank zeros). We use an energy-based approach [34, Section 5.1] with 25 ms, non-overlapping frames to obtain speech/nonspeech labels. This intervention is motivated by reports [28–30] on systematic differences in non-speech segments across bona fide and spoof segments in the ASVspoof 2019 data.

**$\mu$ -law encoding** first applies  $\mu$ -law compression with 255 quantization levels. Then it performs  $\mu$ -law expansion to derive the speech signal affected by quantization error. Note that we do not apply random intervention parameters at the sample level for  $\mu$ -law encoding, unlike other interventions.

We save the perturbed files into 16-bit `.flac` following the original file format of ASVspoof 2019. While the above perturbations are commonly used for different *data augmentations* to improve the generalization of DNNs, here they are used as interventions to purposefully create biased data. This allows us to gauge the extent of shortcut learning taking place in the selected models. While the interventions are applied in asymmetric ways, as described in Table 1, the numbers of training files, test files, and evaluation trials remain as in the original data.

### 3.3. Countermeasures

We consider three different CM models. The first one uses linear frequency cepstral coefficient (LFCC) features with Gaussian mixture model (GMM) [35]. The second one uses LFCCs with light convolutional neural network (LCNN) [36, 37]. Above two systems are being used as baselines for the ASVspoof challenge series<sup>2</sup>. Our third CM is AASIST [38], one of the state-of-the-art systems. It directly operates upon raw waveform input and utilizes graph and graph pooling modules. In this paper, we use a light variant of AASIST with 85K parameters for all experiments.

### 3.4. Mixed Effects Modeling of CM Scores

For the analysis, all scores are normalized with Z-score separately for each configuration and intervention. Then the scores are modeled as defined in (2) where the regression coefficients are estimated using *lme4 package* [32] for R.

## 4. Results

### 4.1. Countermeasure performance

Table 3 shows the comparative performance of the three countermeasure models for the original dataset as well as for var-

<sup>2</sup><https://github.com/asvspoof-challenge/2021>

Table 3: The results applying diverse interventions. Three different countermeasures are trained and tested based on the definition of biased configurations in Table 1.

Intervention	Config.	EER (in %)		
		GMM	LCNN	AASIST
-	<b>O</b>	7.92	1.39	1.39
MP3 compression	<b>A</b>	0.00	0.01	0.01
	<b>B</b>	0.00	0.01	0.01
	<b>C</b>	99.99	94.00	94.00
	<b>D</b>	97.85	99.93	99.93
Additive white noise	<b>A</b>	0.00	0.00	0.00
	<b>B</b>	0.01	0.00	0.00
	<b>C</b>	99.98	99.98	99.98
	<b>D</b>	99.99	99.98	99.98
Loudness normalization	<b>A</b>	7.61	1.60	1.60
	<b>B</b>	7.83	0.66	0.66
	<b>C</b>	8.44	20.66	20.66
	<b>D</b>	9.00	23.86	23.86
Non-speech zeroing	<b>A</b>	2.40	1.06	1.06
	<b>B</b>	0.57	0.38	0.38
	<b>C</b>	81.67	3.52	3.52
	<b>D</b>	90.53	35.67	35.67
$\mu$ -law Quantization	<b>A</b>	0.41	0.80	0.80
	<b>B</b>	0.38	0.25	0.25
	<b>C</b>	78.79	48.54	48.54
	<b>D</b>	82.02	41.98	41.98

ious interventions as explained in Table 1. For the original dataset (configuration **O**), the state-of-the-art DNN-based AASIST method shows the lowest EER whereas the classical GMM classifier shows the worst performance. The configurations with perturbation in the same class across training and test (i.e. **A** and **B**) substantially reduce EER compared to the baseline. This is because the dataset bias acts as an additional cue for discrimination. For some cases, they show extreme case performance with 0% EER indicating perfect discrimination. On the other hand, completely opposite trends are shown when the perturbation is reversed for classes across training and test (i.e., **C** and **D**). In some cases, we observe more than 50% EER indicating label flipping.

In terms of the type of interventions, the models are highly sensitive, particularly in Mp3 compression and additive white noise. The less significant intervention involves loudness normalization through countermeasure models. An important observation is that when intervention is added to spoof utterances in training, it has a greater impact compared to adding intervention to bona fide utterances in training on neural classifiers, shown in the gap between **C** and **D**. We can reasonably conclude that bona fide speech is less susceptible to silence, which aligns with the findings of [28]. Additionally, the trend across different features and models is consistent, indicating a potential data bias.

#### 4.2. Mixed effects modeling of biased scores

We employed a mixed-effect model with Eq. (2) to fit the standardized detection scores and determine the effect of configuration variation on bias across the trials. Each model corresponds to the intervention and its five configurations. Table 4 shows the parameters of each CM model on each intervention. The mean of scores represents the spoofing trials as the reference data with no interventions. We found a substantial effect on  $\beta^*$  (referring

Table 4: Model parameters for countermeasure scores with the tested configurations.  $\mu$  is the model intercept,  $d$  is the class discrimination,  $\beta^*$  refers to the biased training effect in the configurations, where  $\beta^{\text{spf}} = \beta^*$  and  $\beta^{\text{bona}} = -\beta^*$  and  $\varepsilon_i$  is the residual variance.

System	Intervention	$\mu$	$d$	$\beta^*$	$\varepsilon_i$
GMM	MP3	-0.029	0.287	0.513	0.781
	Additive noise	-0.012	0.120	0.533	0.771
	Loudness	-0.083	0.806	0.002	0.939
	Non-speech	-0.025	0.243	0.341	0.901
	$\mu$ -law	-0.056	0.549	0.173	0.948
LCNN	MP3	-0.041	0.402	0.588	0.707
	Additive noise	-0.045	0.443	0.580	0.712
	Loudness	-0.218	2.119	0.010	0.584
	Non-speech	-0.143	1.385	0.237	0.777
	$\mu$ -law	-0.035	0.346	0.475	0.808
AASIST-L	MP3	-0.064	0.623	0.379	0.849
	Additive noise	-0.051	0.501	0.598	0.690
	Loudness	-0.064	0.625	0.008	0.963
	Non-speech	-0.221	2.141	0.089	0.569
	$\mu$ -law	-0.181	1.760	0.236	0.668

to  $\beta^{\text{bona}}$  or  $\beta^{\text{spf}}$ ) for all the models except for loudness normalization in AASIST-L and GMM. This does not mean that a significant effect cannot be found in the individual configurations. However, the models had high residual variances, describing the random effects that could not be explained by our selected variables. This outcome was anticipated due to the experiments' detrimental impact on system performance.

To analyze the effect of each configuration on the detection score, we use the full model parameters for each intervention to define models for each configuration which were presented in Table 2. As outlined in the same Table, differences between trial types were defined by  $d$ , and  $\beta^*$  to compare the configuration effects for spoof or bona fide training. Configurations **A** and **B** exhibited a larger difference between bona fide and spoof trials in estimated scores compared to **O**, resulting in a lower EER. In contrast, configurations **C** and **D** had mismatched training and testing class interventions, leading to a higher EER. In terms of interventions, MP3 compression, and additive white noise showed higher variation effects for  $\beta^*$ , while loudness normalization produced smaller variation effects for the data.

## 5. Conclusions

To uncover shortcut learning in binary classifiers, we propose a novel framework introducing systematic bias with intentional intervention in an asymmetric way. Our goal is to intrigue the black-box model by giving shortcuts explicitly to react to the given conditions. We conduct our proposed method in audio anti-spoofing as a case study. By fitting a mixed-effects model on countermeasure scores from diverse CM models, we demonstrate the effect of data bias on scores due to interventions. The results reveal that MP3 compression and additive white noise could be shortcuts for audio anti-spoofing. Our findings indicate a direction for analyzing possible data biases in countermeasure evaluation. The solutions to mitigate those biases and deep analysis of those correlations remain in our future work.

## 6. Acknowledgements

This work was partially supported by Academy of Finland (Decision No. 349605, project "SPEECHFAKES").

## 7. References

- [1] V. Zue, S. Seneff, and J. Glass, "Speech database development at mit: Timit and beyond," *Speech communication*, vol. 9, no. 4, pp. 351–356, 1990.
- [2] G. R. Doddington, M. A. Przybocki, A. F. Martin, and D. A. Reynolds, "The NIST speaker recognition evaluation—overview, methodology, systems, results, perspective," *Speech communication*, vol. 31, no. 2-3, pp. 225–254, 2000.
- [3] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *Proc. INTERSPEECH*, 2018.
- [4] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, p. 101027, 2020.
- [5] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee *et al.*, "Asvspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, p. 101114, 2020.
- [6] S. Leavy, "Gender bias in artificial intelligence: The need for diversity and gender theory in machine learning," in *Proceedings of the 1st international workshop on gender equality in software engineering*, 2018, pp. 14–16.
- [7] 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.
- [8] C. L. A. Navarro, J. A. Damen, T. Takada, S. W. Nijman, P. Dhiman, J. Ma, G. S. Collins, R. Bajpai, R. D. Riley, K. G. Moons *et al.*, "Risk of bias in studies on prediction models developed using supervised machine learning techniques: systematic review," *bmj*, vol. 375, 2021.
- [9] O. Pfungst, *Clever Hans:(the horse of Mr. Von Osten.) a contribution to experimental animal and human psychology*. Holt, Rinehart and Winston, 1911.
- [10] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann, "Shortcut learning in deep neural networks," *Nature Machine Intelligence*, vol. 2, no. 11, pp. 665–673, 2020.
- [11] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *CVPR*, 2011.
- [12] P. Stock and M. Cisse, "Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases," in *Proc. ECCV*, 2018.
- [13] G. Montavon, W. Samek, and K.-R. Müller, "Methods for interpreting and understanding deep neural networks," *Digital signal processing*, vol. 73, pp. 1–15, 2018.
- [14] B. Kim, H. Kim, K. Kim, S. Kim, and J. Kim, "Learning not to learn: Training deep neural networks with biased data," in *Proc. CVPR*, 2019.
- [15] K. Xiao, L. Engstrom, A. Ilyas, and A. Madry, "Noise or signal: The role of image backgrounds in object recognition," *arXiv preprint arXiv:2006.09994*, 2020.
- [16] D. Teney, E. Abbasnejad, S. Lucey, and A. Van den Hengel, "Evading the simplicity bias: Training a diverse set of models discovers solutions with superior ood generalization," in *Proc. CVPR*, 2022.
- [17] C. J. Anders, L. Weber, D. Neumann, W. Samek, K.-R. Müller, and S. Lapuschkin, "Finding and removing clever hans: using explanation methods to debug and improve deep models," *Information Fusion*, vol. 77, pp. 261–295, 2022.
- [18] S. J. Phillips, M. Dudík, J. Elith, C. H. Graham, A. Lehmann, J. Leathwick, and S. Ferrier, "Sample selection bias and presence-only distribution models: implications for background and pseudo-absence data," *Ecological applications*, vol. 19, no. 1, pp. 181–197, 2009.
- [19] N. McLaughlin, J. M. Del Rincon, and P. Miller, "Data-augmentation for reducing dataset bias in person re-identification," in *International conference on advanced video and signal based surveillance (AVSS)*. IEEE, 2015.
- [20] M. Tian, S. Yi, H. Li, S. Li, X. Zhang, J. Shi, J. Yan, and X. Wang, "Eliminating background-bias for robust person re-identification," in *Proc. CVPR*, 2018.
- [21] W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K.-R. Müller, *Explainable AI: interpreting, explaining and visualizing deep learning*. Springer Nature, 2019, vol. 11700.
- [22] D. Gunning, "Explainable artificial intelligence (xai)," *Defense advanced research projects agency (DARPA), nd Web*, vol. 2, no. 2, p. 1, 2017.
- [23] A. Adadi and M. Berrada, "Peeking inside the black-box: a survey on explainable artificial intelligence (xai)," *IEEE access*, vol. 6, pp. 52 138–52 160, 2018.
- [24] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information fusion*, vol. 58, pp. 82–115, 2020.
- [25] B. Chettri and B. L. Sturm, "A deeper look at gaussian mixture model based anti-spoofing systems," in *Proc. ICASSP*, 2018.
- [26] B. Chettri, S. Mishra, B. L. Sturm, and E. Benetos, "Analysing the predictions of a cnn-based replay spoofing detection system," in *Proc. SLT*, 2018.
- [27] I. Lapidot and J.-F. Bonastre, "Effects of waveform pmf on anti-spoofing detection," in *Proc. INTERSPEECH*, 2019.
- [28] N. M. Müller, F. Dieckmann, P. Czempin, R. Canals, K. Böttinger, and J. Williams, "Speech is silver, silence is golden: What do asvspoof-trained models really learn?" *Proc. INTERSPEECH*, 2021.
- [29] D. Mari, F. Latora, and S. Milani, "The sound of silence: Efficiency of first digit features in synthetic audio detection," in *2022 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2022, pp. 1–6.
- [30] B. Chettri, "The clever Hans effect in voice spoofing detection," in *Proc. SLT*, 2023.
- [31] Y. Zhang<sup>12</sup>, W. Wang<sup>12</sup>, and P. Zhang<sup>12</sup>, "The effect of silence and dual-band fusion in anti-spoofing system," *Proc. INTERSPEECH*, 2021.
- [32] D. Bates, M. Mächler, B. Bolker, and S. Walker, "Fitting linear mixed-effects models using lme4," *Journal of Statistical Software*, vol. 67, no. 1, pp. 1–48, 2015.
- [33] R. ITU-R, "ITU-r bs. 1770-2, algorithms to measure audio programme loudness and true-peak audio level," *International Telecommunications Union, Geneva*, 2011.
- [34] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167639309001289>
- [35] M. Sahidullah, T. Kinnunen, and C. Haniççi, "A comparison of features for synthetic speech detection," in *Proc. INTERSPEECH*, 2015, pp. 2087–2091.
- [36] G. Lavrentyeva, S. Novoselov, E. Malykh, A. Kozlov, O. Kudashov, and V. Shchemelinin, "Audio replay attack detection with deep learning frameworks," in *Proc. INTERSPEECH*, 2017, pp. 82–86.
- [37] X. Wang and J. Yamagishi, "A comparative study on recent neural spoofing countermeasures for synthetic speech detection," in *Proc. INTERSPEECH*, 2021, pp. 4259–4263.
- [38] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, and N. Evans, "Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks," in *Proc. ICASSP*, 2022.