Protecting Publicly Available Data With Machine Learning Shortcuts

Nicolas M. Müller[1]
nicolas.mueller@aisec.fraunhofer.de
Maximilian Burgert[2]
max.burgert@tum.de
Pascal Debus[1]
pascal.debus@aisec.fraunhofer.de
Jennifer Williams[3]
j.williams@soton.ac.uk
Philip Sperl[1]
philip.sperl@aisec.fraunhofer.de
Konstantin Böttinger[1]
konstantin.boettinger@aisec.fraunhofer.de

[1] Fraunhofer AISEC
Lichtenbergstraße 11
85748 Garching, Germany

[2] TU Munich
Arcisstraße 21
80333 München, Germany

[3] University of Southampton
University Road
Southampton, UK

Abstract

Machine-learning (ML) shortcuts or spurious correlations are artifacts in datasets that lead to very good training and test performance but severely limit the model's generalization capability. Such shortcuts are insidious because they go unnoticed due to good in-domain test performance. In this paper, we explore the influence of different shortcuts and show that even simple shortcuts are difficult to detect by explainable AI methods. We then exploit this fact and design an approach to defend online databases against crawlers: providers such as dating platforms, clothing manufacturers, or used car dealers have to deal with a professionalized crawling industry that grabs and resells data points on a large scale. We show that a deterrent can be created by deliberately adding ML shortcuts. Such augmented datasets are then unusable for ML use cases, which deters crawlers and the unauthorized use of data from the internet. Using real-world data from three use cases, we show that the proposed approach renders such collected data unusable, while the shortcut is at the same time difficult to notice in human perception. Thus, our proposed approach can serve as a proactive protection against illegitimate data crawling.

1 Introduction

Machine learning shortcuts or spurious correlations are artefacts in data that significantly change the learning process of models. These features F contain no real semantic information, but have a strong correlation with a target label L nevertheless, i.e. $P(L|F) \neq P(L)$. For example, in audio data the presence or absence of leading silence in speech recordings correlates strongly with whether the corresponding audio is real or a deepfake. Synthesized

© 2023. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

speech recordings often have no or very little leading silence due to text-to-speech (TTS) data processing. Models take advantage of this and classify according to the length of the leading silence [24]. In vision research such as X-ray image datasets for the detection of Covid-19, the label 'sick/healthy' correlates with the type of X-ray equipment used. Learning models thus do not learn to distinguish between sick and healthy patients, but merely to distinguish between X-ray machines [7]. This makes the model useless in practice, c.f. Figure 1.

The challenge in dealing with ML shortcuts is that practitioners often are not aware of their presence. This is because even with a valid train/test split, it is hard to notice that the model is not generalizing. Due to errors in the data collection process, shortcuts are also present in the test data, which results in good testing performance. It seems that new data unseen during training is adequately handled. It is therefore essential to understand whether the model learns shortcuts or actually semantically significant features.

However, ML shortcuts can also be used productively, as we show in this paper. The ability to render datasets unusable for machine learning can be used to protect publicly available, yet proprietary datasets. Many companies offer access to labelled data via websites, apps, or APIs. Used vehicle dealers such as `cars.com` or AutoScout publish ads for used vehicles on their websites and include labels such as vehicle type, make, age, mileage, etc. Dating platforms like Tinder, Bumble and co. publish photos of users incl. description text and labels such as nationality, sexual preference, gender, hometown, ethnicity, and place of residence. Furthermore, clothing manufacturers like Zalando or Esprit publish large catalogues of clothing items on their websites, labelled by category, colour, and price.

All of this data is potentially interesting for machine learning, and a large number of vendors sell crawling services to collect this data, process it, and make it usable for ML. In the process, the circumvention of protection measures is also explicitly advertised. This industry has a yearly turnover of USD $402 million [14] and may harm legitimate data creators. Their intellectual property is violated, and their infrastructure is overloaded or even damaged by crawling. Additionally, in the case of user information, highly sensitive personal data flows into third-party hands.

We propose to protect such datasets by ML shortcuts that render the data unusable by ML, making crawling unattractive and thereby protecting the producers and users, as well as leaving the visual presentation of the data unaffected so that vendors can continue to serve it as usual. This paper presents the following contributions:

- We evaluate the impact of several visual shortcuts on different datasets and show that the generalization ability of the models decreases by over 50%.
- We evaluate Explainable-AI methods for shortcut detection.
- We introduce ML shortcuts as a novel protection measure for public datasets. Using real-world use cases and data, we show that cleverly employed shortcuts can render public datasets unusable such that there is a strong disincentive for illegitimate crawling.

2 Related Work

Machine learning shortcuts have not yet entered the wider consciousness of the scientific community, but academic research on them does exist. One of the first shortcuts in the literature was found in the Pascal VOC 2007 dataset: Here, all images of horses had a watermark

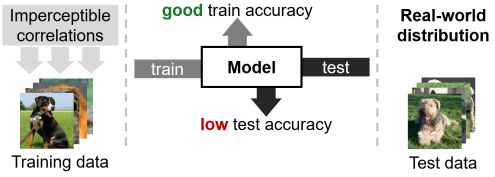


Figure 1: Visualisation of the impact of shortcuts on the machine learning pipeline: Errors in the data-collection process create imperceptible correlations between data and target, which lead to good train accuracy, but do not transfer to the real-world distribution.

of the photographer in the lower right corner. The model then learned to identify horses based on this watermark alone[19]. Similarly, a recent Nature publication [7] looks at the generalization ability of Covid-19 detection algorithms. The authors investigate why such algorithms do not generalize and conclude that the models mainly identify shortcuts such as patient position, the presence of tubes and other medical equipment, or the type of X-ray machine itself. All of this allows conclusions about Covid-19 within the data set, but does not generalize.

Shortcuts also occur in the classification of audio deepfakes. The most established data set in this area contains a shortcut in which the label correlates with the length of the leading silence. If the silence is removed, the model performance deteriorates by up to a factor of five [24]. Recently published work proposes several approaches to discovering or even removing these shortcuts from the dataset. However, this remains a challenging problem even with precise knowledge of the dataset [9]. ML shortcuts do not only affect classification: self-supervised methods like contrastive learning are also vulnerable [23, 27].

Recent work also highlights the use of ML shortcuts to protect personal data. Using shortcuts as 'machine learning availability attacks', the authors of [37] show that their effectiveness lies in the linear separability of shortcuts and data. The authors of [13] create 'unlearnable examples' by crafting an error-minimizing noise that tricks the model to learn nothing from a given data point. This, however, is based on adversarial perturbations and requires white-box access to an attacker's assumed learning model. Alternatively, personal data can be protected using data poisoning [8, 32]. Unlike shortcuts, however, which are model-agnostic, adversarial perturbations require the target model architecture and/or weights, since the perturbation δ is found via gradient-based techniques.

Finally, our approach of ML shortcuts has similarities to the field of digital watermarking, where data is covertly embedded in a carrier signal in order to enforce usage control, e.g. with respect to copyright of audio or video content. Likewise, ML shortcuts also aim for usage control in the sense that usage for third-party machine learning applications becomes technically infeasible. Whereas watermarks restrict unauthorized presentation of data (such as images on websites), ML shortcuts restrict unauthorized usage in knowledge discovery and machine learning.

3 Machine Learning Shortcuts

In this section, we present ML shortcuts based on plausible data-collection errors and investigate their impact on known image classification problems. Additionally, we evaluate the applicability of different Explainable-AI (XAI) techniques for shortcut detection.

(a) Original (b) Dust (c) Hue (d) Lens (e) Sensor

Figure 2: Visualisation of the used shortcuts, which correspond to real-world data collection errors: dust on the lens, differences in ambient light (Hue), impairment of the photo lens (Lens), or sensor error (Sensor). Top row: ImageNette, Bottom Row: Visualisation of the shortcut on a constant background.

3.1 ML-Shortcuts Due To Data Collection Errors

We discuss four different types of shortcuts which reflect real-world data collection errors, and which can be leveraged for protecting publicly available data. First, the obstruction of a few image pixels by dust particles may inadvertently encode the class label (*dust shotcut*). This mimics dust on the camera lens when collecting real-world data of a particular class. Second, slight overall color changes in an image can indicate the class (*hue shortcut*). This may be, for example, due to different weather conditions when collecting the images, for example when collecting images of class A in the morning and images of class B in the evening. Third, specific camera settings may cause alterations to the border of the image, resulting in a lens-like effect that may hint at the target label (*lens shortcut*). This might result from different settings when taking pictures from different classes, for example when using different levels of zoom or exposure. Fourth, specific low-intensity color patterns stemming from, for example, a characteristic or faulty camera sensor can indicate the image class (*sensor shortcut*) [38]. Here, the assumption is that different classes were collected using different cameras. We present examples of the shortcuts in Figure 2 and evaluate how they impact classification models in Section 5.2.

3.2 Explainable AI for Shortcut Detection

One obvious approach for the mitigation of ML-Shortcuts is Explainable AI (XAI). When it is clearly understood what the model is learning, mitigation strategies can be derived. Shortcuts may either be removed manually, for example by adequately cropping or post-processing the input. Alternatively, new data can be collected and applied as a shortcut. And finally, efforts can be made to counteract the shortcut, for example using segmentation masks [36]. Our goal is to evaluate whether and to what degree XAI methods can indeed detect shortcuts.

3.2.1 Explainable AI Overview

Explainable AI strives to make the behavior of a learning model f explainable: for some input x, XAI-methods λ typically produce an explanation $z = \lambda(f(x))$ of the same dimensionality as the image. From the large number of available XAI methods [20], we limit our analysis to some of the most popular methods:

Saliency maps (SM) [33] calculate the magnitude of the gradient with respect to the loss function for some input x. This results in a heatmap z, which shows the influence of single pixels and regions on the final classification result. Closely related is the Integrated Gradients Methods (IG) [35], where the model's gradients are computed for a progression of interpolations of a baseline and the input image. Similarly, Smooth Grad [34] computes regions of interest by analyzing the model's gradients w.r.t. the input image but uses additive gaussian noise in order to create averaged explanations. This reduces the noise in the resulting explanations. Finally, Grad-Cam (GC) [31], a refinement of Class Activation Map (CAM, [39]), analyzes the gradient of the model's prediction $f(x)$ in order to compute the averages of the penultimate convolutional layers, which allows identifying which parts of an input image contribute to the model prediction the most.

3.2.2 Evaluation Strategy

We apply all of these techniques to our datasets and models and investigate whether the presented shortcuts can be detected. To this end, we train models on shortcut-affected datasets and compare the XAI representation $\lambda(f(x))$ with that of a model trained on a clean dataset. For saliency maps, for example, we compare the absolute gradients between shortcut and clean models via L_2, using the same input image in each case. Formally:

$$\frac{1}{N}\sqrt{\sum_{i=1}^{N}\Big(\lambda(f_\theta(x_i),y_i)-\lambda(f_\gamma(x_i),y_i)\Big)^2} \tag{1}$$

where x_i, y_i represent the data in a dataset of size N. λ is some Explainable AI method

$$\lambda : \mathbb{R}^K \times \mathbb{N} \to \mathbb{R}^K \tag{2}$$

$$x_i, y_i \mapsto z_i \tag{3}$$

and θ and γ are clean and shortcut-affected model parameters, respectively. The larger this L_2 difference, the more the shortcut is potentially identifiable by the corresponding XAI method. Section 5.3 presents the results.

4 Using Shortcuts to Protect Data

Despite all the challenges they introduce, ML-Shortcuts can also be used beneficially, protecting public but proprietary datasets. This is because shortcuts can be used to discourage web scraping. The profitable but problematic collection of proprietary data from open-access sources such as websites and apps.

4.1 The Threat of Web Scraping

The market for web scraping generated USD $402 million in revenue in 2020 [14] and is expected to surpass USD $1.7 billion in 2030 [22]. Numerous vendors offer web scraping explicitly for the creation of machine learning datasets [5, 28, 29].

Observers estimate that web scraping causes millions of dollars in damage, and up to two percent of web store sales lost [4]. Recent research [16, 17] has shown that besides the economic effects of web scraping, there are also legal and ethical implications.

Not only is the scraped data often a critical and proprietary asset of the targeted website but the scraping process itself puts a strain on the infrastructure potentially compromising the availability of the service provided by the website. In US law, the latter aspect is a tort that is also known as *trespass to chattels* and led to a number of court cases, such as eBay versus Bidder's Edge (2000) [6]. Additional web scraping lawsuits are discussed in [16] or [26]. Other important legal aspects are a violation of copyrights and confidentiality or are concerned with how the scraped data is used and of course, the access to the scraped data might have violated the terms of service or might have been illegal in the first place.

From an ethical perspective, web scraping compromises confidentiality and privacy of the users of a website (consider, e.g., the cases of scraping data from a dating platform) and, depending on how the leaked data is used, might contribute to bias and discrimination.

Legislation such as the Computer Fraud and Abuse Act [1], the Digital Millennium Copyright Act [2], or the European General Data Protection Regulation [3], provide defense or at least compensation mechanisms against scrapers, however, as shown by [30], legal professionals are struggling to define precisely what web scraping actually means resulting in some oscillations between too broad and too narrow interpretations over the last two decades.

Technical measures such as *captchas*, obfuscated HTML code, or access restrictions can increase the effort required by scrapers. However, this is associated with significant effort on the defender side, and can still be circumvented by the scrapers. This 'arms race' is asymmetrical and to the advantage of the scraper, whose very core business it represents as opposed to the defender. A different kind of defense is therefore necessary.

4.2 Technical Description of Proposed Defense

We suggest data owners add shortcuts to proprietary, publicly available, implicitly labeled data such that it is no longer an attractive target for crawling and subsequent machine learning use. If data is labeled with respect to several categories, e.g. pictures from dating platforms according to gender, religious affiliation as well as ethnicity, each combination of labels must be encoded by the shortcut. Since this complexity increases rapidly, the following requirements for shortcuts arise. First, to be able to encode as large a number of labels as possible. Second, to strongly influence the training of ML models so that they extract as little information as possible from the original data. Third, to be as inconspicuous as possible for human perception.

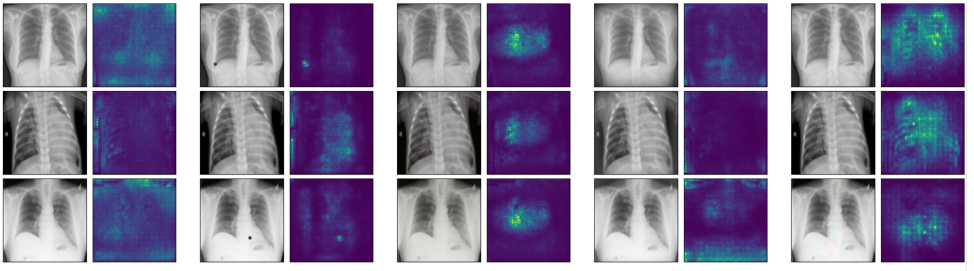
5 Experiments and Evaluation

5.1 Data and Methodology

We evaluate our proposed shortcuts (c.f. Figure 2) on four image classification tasks: Imagenette [11], a subset of the ImageNet Dataset, Covid-QU-Ex [36], CIFAR10 and CIFAR100 [15]. We use a pre-trained DenseNet-121 [12], which we fine-tune on all of the datasets. We train our models using PyTorch [25], using a learning rate of 0.001, a batch size of 256, and data augmentation (centre crop, vertical flip and random translations). The shortcuts are added *before* the data augmentation, as would be the case in the real world. We then train for 40 epochs (using early stopping) and report accuracy on a separate test set

	original	hue	lens	dust	sensor
Covid	93.7 ± 0.6	34.1 ± 1.3	35.0 ± 0.0	35.8 ± 2.7	33.7 ± 1.5
ImageNette	87.2 ± 0.4	47.5 ± 0.8	84.2 ± 0.2	88.1 ± 0.5	40.4 ± 5.4
CIFAR10	90.0 ± 0.4	47.9 ± 0.5	48.2 ± 0.3	78.5 ± 1.6	51.1 ± 0.4
CIFAR100	68.4 ± 0.3	35.4 ± 0.0	30.1 ± 0.7	54.5 ± 1.8	38.4 ± 1.0

Table 1: The impact of machine-learning shortcuts on DenseNet-121 on the CIFAR10, CIFAR100 and ImageNette. Accuracy aggregated over three independent trials, with standard deviation shown.



(a) Baseline (b) Dust Shortcut (c) Hue Shortcut (d) Lens Shortcut (e) Sensor Shortcut

Figure 3: Visualisation of smooth-grad output for all shortcuts presented.

(about 10% the size of the training data). The training data either has no shortcut (*original*) or one of the four shortcuts mentioned above.

5.2 Impact of ML Shortcuts

As shown in Table 1, the shortcuts strongly deteriorate model performance, from about 87% to 42% in test accuracy for ImageNette and from about 93% to 33% for Covid-QU-Ex, which is equivalent to random guessing for a three-way classification problem. We can observe that while the shortcuts are effective for all datasets, they are especially powerful when the classification task is hard as in the case of Covid-QU-Ex.

5.3 XAI for shortcut detection

We now analyze whether XAI methods can detect such shortcuts. Consider Table 2, which provides the XAI-score as derived by Equation (1), aggregated over all models and datasets. We compute the difference in Explainable AI output between a baseline model, trained on a non-shortcut dataset, and one of the following. First, a control model, which is also trained on a non-shortcut dataset. This is in order to have a comparison of how the XAI output differs between two identically trained models. Second, a shortcut-affected model, where we use one of the four shortcuts introduced in Section 3.1.

We can see that even though the baseline and control models are identically set up, they have different XAI outputs. The shortcut-affected models however have an even larger L_2 difference in XAI output. Consider for example the Sensor shortcut, which reduced the model performance on ImageNette from 87% to 40%, c.f. Table 1. For the Smooth Grad (SG) Explainable-AI method, the control has a difference of 7.5 to the baseline, while the

	control	dust	hue	lens	sensor
SG	7.5 ± 2.3	9.1 ± 2.4	8.4 ± 2.4	8.1 ± 2.3	15.9 ± 1.9
SM	9.7 ± 2.5	11.7 ± 2.7	10.8 ± 2.7	10.5 ± 2.6	11.2 ± 2.7
IG	24.8 ± 7.8	24.4 ± 7.8	24.9 ± 7.8	25.0 ± 7.7	40.2 ± 9.7
GC	0.7 ± 0.4	1.5 ± 0.5	0.8 ± 0.4	0.8 ± 0.4	2.6 ± 0.3

Table 2: The L_2 difference in four XAI output for clean and shortcut-affected models on the COVID dataset, c.f. Equation (1). Higher values indicate that the addition of the shortcut triggers different XAI output, meaning that the shortcut has a higher chance of being detected by humans.

Sensor shortcut has a difference of 15.9. This means that, for the most part, the shortcut changes the XAI output dramatically, which should allow identification by XAI methods.

This is corroborated by Figure 3. The model learns to ignore the regions of interest in the original training data and only focuses on the shortcut-affected areas. Small, pixelated areas for the dust shortcut, the outer regions of the image for the Lens shortcut, or specific checkerboard-like patterns for the sensor shortcut.

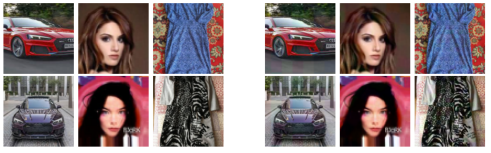
5.4 Protection against web-scraping

To leverage ML-Shortcuts beneficially, we suggest data owners employ shortcuts to make real-world datasets unlearnable. Based on Table 1, we suggest using the sensor shortcut. This is because it is highly effective, cannot easily be removed (as, for example, the dust shortcut), but also visually nearly imperceptible.

We evaluate our approach not on the datasets proposed in Section 5.1, but on the three real-world use cases: the protection of image data from dating platforms, used car dealers, and the fashion industry.

- Online Dating. We use the CelebA data set [21], which contains 202,599 face images of celebrities, and is annotated with 40 binary attributes. We select five particularly sensitive binary attributes. Attractive, Male, Young, Pale_Skin, Bald. The defender's shortcut must therefore cover $2^5 = 32$ combinations of features.
- Fashion. We use the Clothing dataset [10], which consists of 5000 real-world images of 20 different clothing items.
- Used Cars. We use the Cars Dataset [13], which contains 4000 images of seven car types.
- Additionally, we obtained permission to collect a dataset of real-world used car images from a major European online car vendor. For 10 different car models, we collect 800 images each, which results in a dataset of 8000 images of used cars people uploaded in late 2022.

For each of these datasets, we create a sensor shortcut (c.f. Section 3) and add it to all the data points in the training set. We then train different models on this training set, and then evaluate these models on the unperturbed test dataset. This procedure serves as a proxy to estimate the real-world generalization ability of the model trained on the shortcut dataset. For CelebA, we create only one shortcut, which is then used to protect five different attributes.



(a) No Shortcut (b) Sensor Shortcut

Figure 4: Samples with and without the sensor shortcut, for the three publicly available datasets Cars, CelebA and Clothing.

	original	sensor
Online Dating (32 classes)	67.9 ± 0.0	7.5 ± 2.7
Fashion (20 classes)	87.7 ± 4.5	34.1 ± 6.6
Used Cars (7 classes)	60.6 ± 0.1	34.7 ± 11.8
Used Cars (real world, 10 classes)	94.8 ± 0.2	15.2 ± 7.4

Table 3: Test accuracy of a Dense Net, trained either on the original dataset, as well as the sensor-shortcut affected dataset. The introduction of the shortcut degrades the model, making it unsuited for productive use.

This corresponds to a real-world system where a defender does not know the attacker's use case.

Figure 4 visualizes the application of our proposed sensor shortcut. Additionally, Table 3 presents the results of training a deep neural network (DenseNet-121, c.f. Section 5.1) on this data. It can be seen that in each case, the performance of the model is significantly reduced so that productive use of the model would no longer be possible. For example, for the real-world Used Car dataset, the performance is reduced from 94.8 to 15.2 percent. At the same time, the shortcut is difficult to perceive visually due to the small magnitude of the perturbation added, c.f. Figure 4. This will allow data vendors to still openly employ the data in question while crawling for machine learning usage is strongly disincentivized.

6 Conclusion

We show that shortcuts can greatly affect the performance of models. This enables practical use cases in protecting publicly available but proprietary data, such as implicitly labelled datasets in fashion or used vehicles. Since they cannot be detected with the analysis of train/test performance, the use of Explainable AI methods is necessary. As we show, these methods can indicate the presence of shortcuts. In future work, explainable AI methods might serve to attack ML shortcuts, namely to identify them in order to remove them automatically. However, cleaning datasets from ML shortcuts without creating new shortcuts and artefacts is a challenging question on its own and subject to future research.

References

[1] Computer fraud and abuse act. *Legislation passed at the 99th Congress, 2nd session*, 1986.

[2] Digital millennium copyright act. *Public Law*, 105(304):112, 1998.

[3] 2018 reform of eu data protection rules, 2018. URL https://ec.europa.eu/commission/sites/beta-political/files/data-protection-factsheet-changes_en.pdf.

[4] Eleanor Ajala. The economics of web scraping report | imperva. https://www.imperva.com/blog/the-economics-of-web-scraping-report/, 8 2018. (Accessed on 10/20/2022).

[5] Bright Data Ltd. Bright data - the world's #1 web data platform. https://brightdata.com/. (Accessed on 10/20/2022).

[6] Edward W Chang. Bidding on trespass: ebay, inc. v. bidder's edge, inc. and the abuse of trespass theory in cyberspace-law. *AIPLA QJ*, 29:445, 2001.

[7] Alex J DeGrave, Joseph D Janizek, and Su-In Lee. Ai for radiographic covid-19 detection selects shortcuts over signal. *Nature Machine Intelligence*, 3(7):610–619, 2021.

[8] Liam Fowl, Ping-yeh Chiang, Micah Goldblum, Jonas Geiping, Arpit Bansal, Wojtek Czaja, and Tom Goldstein. Preventing unauthorized use of proprietary data: Poisoning for secure dataset release. *arXiv preprint arXiv:2103.02683*, 2021.

[9] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[10] Alexey Grigorev. Clothing dataset (full, high resolution) | kaggle. https://www.kaggle.com/datasets/agrigorev/clothing-dataset-full, 02 2022. (Accessed on 12/02/2022).

[11] Jeremy Howard. fastai/imagenette: A smaller subset of 10 easily classified classes from imagenet, and a little more french. https://github.com/fastai/imagenette, 12 2019. (Accessed on 12/02/2022).

[12] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[13] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable. *arXiv preprint arXiv:2101.04898*, 2021.

[14] Sagar Kadam. Current research: Web scraper software market analysis, regional analysis 2020: 2027. https://www.whatech.com/og/markets-research/it/710912-web-scraper-software-market-analysis-regional-analysis-2020-2027, 8 2021. (Accessed on 10/20/2022).

[15] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[16] Vlad Krotov and Leiser Silva. Legality and ethics of web scraping. In *24th Americas Conference on Information Systems, AMCIS 2018, New Orleans, LA, USA, August 16-18, 2018*. Association for Information Systems, 2018. URL https://aisel.aisnet.org/amcis2018/DataScience/Presentations/17.

[17] Vlad Krotov, Leigh Johnson, and Leiser Silva. Tutorial: Legality and ethics of web scraping. *Commun. Assoc. Inf. Syst.*, 47:22, 2020. doi: 10.17705/1cais.04724. URL https://doi.org/10.17705/1cais.04724.

[18] Kshitij Kumar. Car images dataset | kaggle. https://www.kaggle.com/datasets/kshitij192/cars-image-dataset, 05 2022. (Accessed on 12/02/2022).

[19] Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature Communications*, 10, 03 2019. doi: 10.1038/s41467-019-08987-4.

[20] Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1):18, 2020.

[21] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[22] Market Research Future. Web scraper software market segment, size, share, global trends, 2030 | mrfr. https://www.marketresearchfuture.com/reports/web-scraper-software-market-10347#, 11 2020. (Accessed on 10/20/2022).

[23] Matthias Minderer, Olivier Bachem, Neil Houlsby, and Michael Tschannen. Automatic shortcut removal for self-supervised representation learning. In *International Conference on Machine Learning*, pages 6927–6937. PMLR, 2020.

[24] Nicolas M Müller, Franziska Dieckmann, Pavel Czempin, Roman Canals, Konstantin Böttinger, and Jennifer Williams. Speech is silver, silence is golden: What do asvspoof-trained models really learn? *arXiv preprint arXiv:2106.12914*, 2021.

[25] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[26] Kathleen C Riley. Data scraping as a cause of action: limiting use of the cfaa and trespass in online copying cases. *Fordham Intell. Prop. Media & Ent. LJ*, 29:245, 2018.

[27] Joshua Robinson, Li Sun, Ke Yu, Kayhan Batmanghelich, Stefanie Jegelka, and Suvrit Sra. Can contrastive learning avoid shortcut solutions? *Advances in neural information processing systems*, 34:4974–4986, 2021.

[28] ScrapeHero. Training data for artificial intelligence and machine learning - scrapehero. https://www.scrapehero.com/machine-learning-ai-training-data/. (Accessed on 10/20/2022).

[29] ScrapeStorm. Ai-powered visual web scraping tool | scrapestorm. https://www.scrapestorm.com/. (Accessed on 10/20/2022).

[30] Andrew Sellars. Twenty years of web scraping and the computer fraud and abuse act. *BUJ Sci. & Tech. L.*, 24:372, 2018.

[31] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[32] Juncheng Shen, Xiaolei Zhu, and De Ma. Tensorclog: An imperceptible poisoning attack on deep neural network applications. *IEEE Access*, 7:41498–41506, 2019.

[33] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Workshop at International Conference on Learning Representations*, 2014.

[34] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[35] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[36] Anas M. Tahir, Muhammad E. H. Chowdhury, Yazan Qiblawey, Amith Khandakar, Tawsifur Rahman, Serkan Kiranyaz, Uzair Khurshid, Nabil Ibtehaz, Sakib Mahmud, and Maymouna Ezeddin. Covid-qu-ex dataset, 2022. URL https://www.kaggle.com/dsv/3122958.

[37] Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. Availability attacks create shortcuts. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2367–2376, 2022.

[38] Aidong Zhang, Huzefa Rangwala, Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. Availability Attacks Create Shortcuts. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2367–2376, 2022. doi: 10.1145/3534678.3539241.

[39] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.