ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10446824

DETECTION AND ATTRIBUTION OF MODELS TRAINED ON GENERATED DATA

Ge Han[1], *Ahmed Salem*[2], *Zheng Li*[3*], *Shanqing Guo*[4*†], *Michael Backes*[3], *Yang Zhang*[3]

[1] School of Computer Science and Technology, Shandong University
[2] Azure Research [3] CISPA Helmholtz Center for Information Security
[4] School of Cyber Science and Technology, Shandong University

ABSTRACT

Generative Adversarial Networks (GANs) have become widely used in model training, as they can improve performance and/or protect sensitive information by generating data. However, this also raises potential risks, as malicious GANs may compromise or sabotage models by poisoning their training data. Therefore, it is important to verify the origin of a model's training data for accountability purposes. In this work, we take the first step in the forensic analysis of models trained on GAN-generated data. Specifically, we first detect whether a model is trained on GAN-generated or real data. We then attribute these models, trained on GAN-generated data, to their respective source GANs. We conduct extensive experiments on three datasets, using four popular GAN architectures and four common model architectures. Empirical results show the remarkable performance of our detection and attribution methods. Furthermore, we conduct a more in-depth study and reveal that models trained on various data sources exhibit different decision boundaries and behaviours.

Index Terms— Generative Adversarial Networks (GANs), GAN-trained models, forensic analysis, accountability

1. INTRODUCTION

Deep learning has become a vital tool for various domains, such as business, healthcare, industries, and military [1]. Its rapid growth and wide-ranging applications depend on the availability of abundant data [2]. However, data collected in the real world are small, dirty, biased and even poisoned [3]. Furthermore, data collection and usage are subject to legal restrictions such as GDPR [4] and DPIA [5]. These issues have motivated research on generative methods, especially Generative Adversarial Networks (GANs). GANs can learn the distributions of target datasets and generate new data. Therefore, they can overcome the limitations of scarce training data. Notably, leading companies like Google, Intel, and NVIDIA have adopted synthetic data from providers such as MostlyAI [6], Datagen [7], YData [8], and Bitext [9] to train their deep learning models.

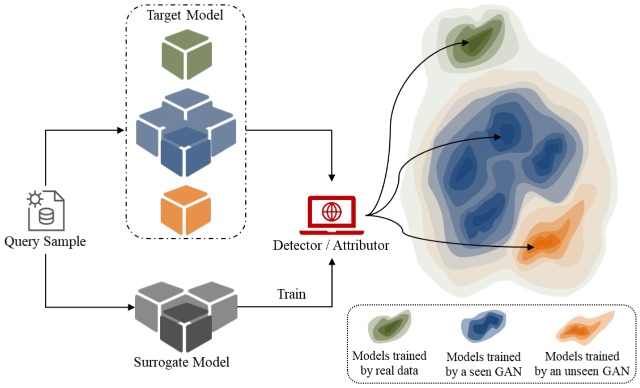

Target Model
Query Sample
Detector / Attributor
Train
Surrogate Model
Models trained by real data
Models trained by a seen GAN
Models trained by an unseen GAN

Fig. 1: An illustration of our work.

Using GAN-generated data to train models can introduce serious and realistic threats. Malicious GANs can create harmful data, either passively or actively. Models trained on such data can pose accountability risks for the owner, as they may be involved in malicious activities related to their models. For example, GANs can generate poisoning data that degrades model robustness, leading to significant performance drop [10, 11]. Moreover, GAN-generated data can embed backdoors that can trigger unwanted model outputs [12]. Several forensic techniques can trace the source of generated data, addressing the concerns about its quality [13, 14, 15, 16, 17, 18]. However, these techniques only work at the data level. No previous work focuses on the model level, i.e., the models trained on GAN-generated data.

Contributions. This paper fills this gap by conducting the first study on detecting and attributing models trained on GAN-generated data, as shown in Fig. 1. Specifically, we aim to answer the following research questions (**RQ**s):

- **RQ1.** Can we distinguish models trained on GAN-generated data from those trained on real data, i.e., can

[*] Corresponding authors.

[†] The work is partially supported by the National Natural Science Foundation of China under Grant (No. 62372268), Shandong Provincial Natural Science Foundation (No. ZR2021LZH007, No. ZR2022LZH013), Funding Project of Jianghuai Advance Technology Center (No. 00QK0021) and Jinan City "20 New Universities" Funding Project (2021GXRC084).

we detect GAN-trained models?

- **RQ2.** Can we link models trained on generated data to the source GANs that created their training data, i.e., can we attribute these models to their training data sources?

To address the distinction between models trained on GAN-generated and real data (**RQ1**), we construct a binary classifier, termed the detector. We initiate this process by training surrogate models separately using data generated by multiple GANs alongside real data. Subsequently, we input a fixed probing set into each surrogate model and label the resultant output as either "real" or "GAN." We combine all labelled model outputs into a dataset that functions as the training set for the detector. Following training, the detector can ascertain whether a given target model is trained on GAN-generated data.

Upon identifying a model as being trained on GAN-generated data, we proceed to attribute it to the specific GAN accountable for generating its training data (**RQ2**). We explore two scenarios: closed-world attribution and open-world attribution. In the former, the goal involves pinpointing the origin of the target model's training data from a finite set of potential GANs. To achieve this, we construct an attributor functioning as a multi-class classifier. This attributor employs the target model's output to predict the GAN responsible for the data. Conversely, in the open-world scenario, we relax the assumption that the attributor has access to all potential GANs. In this case, we introduce a binary classifier as the attributor. This classifier is designed to establish the relationship between the target model and each suspected GAN, offering enhanced flexibility.

Our investigation encompasses extensive experiments across three image datasets, four prominent GANs, and four widely-used CNN architectures[1]. Empirical findings underscore the impressive performance of both our detector and attributor.

2. GAN-TRAINED MODEL DETECTION

In this section, we present our detection for whether models have been trained on GAN-generated data or not (**RQ1**).

2.1. Design Goals

To tackle the threats posed by training models on GAN-generated data, the design of our detector should effectively distinguish between models trained on generated data and those on real data. We refer to these models as "Target Models."

2.2. Methodology

To this end, we build a detector by training a binary classifier. The design of our detector encompasses three stages: Surrogate Model Construction, Dataset Construction, and Detector Construction.

- **Surrogate Model Construction.** Initially, the real dataset is partitioned into three parts: training set, testing set, and probing set. We proceed by training multiple GAN models using diverse subsets randomly sampled from the training set. This results in a collection of data sources, comprising GAN-generated data and the original training set. For each source, we train numerous classification models, referred to as surrogate models.

- **Detector Dataset Construction.** We use the probing set to query these surrogate models and label the outputs of models trained on GAN-generated/real data as 0/1. Subsequently, we build a binary dataset tailored to the detectors.

- **Detector Construction.** Employing classical training techniques, we train the detector from scratch using the binary dataset.

Once the detector has been trained, we subject the testing set to the above second stage to feed the probing set to the target models trained from other GANs or real data to obtain an independent binary dataset. Finally, we evaluate the generalizability of the detector using this new binary dataset.

2.3. Experimental Setup

Datasets and GAN Architectures. We consider three classification image datasets: CelebA [19], Fashion-MNIST (FM-NIST) [20], and SVHN [21]. Table 1 shows the dataset partition setting. We consider four popular conditional GAN architectures, namely, CGAN [22], DCGAN [23], ACGAN [24], and WGAN [25]. The reason is that they can generate images based on corresponding labels, enabling training classification models in a supervised way. Furthermore, we train 40 GAN models, with each GAN architecture contributing 10 instances.

Target/Surrogate Model Architecture. We use ResNet-18, a representative CNN model, as the target model. Since we assume we only have black-box access to the target model, we use a different CNN model, VGG-9, as the surrogate model. To evaluate GAN-trained model detection, we divide the real training set into two equal portions. One portion is utilized alongside 20 GANs to train surrogate models and create the dataset for training the detector. Simultaneously, the other half of the real training set, in conjunction with the other 20 GANs, is employed to train models acting as target models,

[1] https://github.com/G3H4N/GAN-Trained-Model-Detection-and-Attribution

Table 1: The Dataset Partition Setting.

Dataset	Training Set	Testing Set	Probing Set	Subset
CelebA	192499	10000	100	100000
FMNIST	59900	10000	100	50000
SHVN	73257	25932	100	70000

Table 2: The Accuracy of Model Detection.

Dataset	Detector I	Detector II	Detector III
CelebA	0.705	**0.948**	0.933
FMNIST	0.691	**0.926**	0.850
SVHN	0.689	**0.942**	0.915

forming the dataset for testing the detector. For each source, we train 20 classifiers as the surrogate/target models.

Detector. We construct the detector as a 2-layer MLP, where the input dimension aligns with the binary dataset's data size, and the output layer's dimension is set at 2. We consider three detectors, namely Detector I, II, and III. For each detector, we construct the binary dataset with different information related to model output, including posterior alone for Detector I, posterior along with an additional 1/0 label indicating prediction correctness for Detector II, and posterior combined with labels from the probing set for Detector III.

2.4. Results

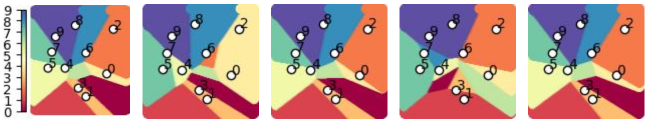
We present the detection accuracy in Table 2. Notably, our detectors demonstrate impressive performance. The highest accuracy is achieved on CelebA, reaching up to 94.8%, while the lowest is observed for SVHN, still exceeding 68.9%. These results verify that models trained on GANs exhibit distinct behaviours when compared to those trained on real data. In essence, these models are capable of revealing information about their training data.

Moreover, Detector II attains the highest accuracy. This observation suggests that beyond the posterior, the additional 1/0 label indicating prediction correctness can further enhance the detection performance. The reason is that the correctness judgement leaks more information involving model behaviour for detection.

3. GAN-TRAINED MODEL ATTRIBUTION

3.1. Design Goals

The primary goal of GAN-trained model attribution is to effectively attribute different models trained on GAN-generated data to their source GANs. The purpose of attribution is to hold the GAN owners accountable for the possible misbehaviour of the models trained on their generated data.



(a) Real Data (b) CGAN (c) ACGAN (d) DCGAN (e) WGAN

Fig. 2: The visualization of decision boundaries for VGG-9 classifiers trained from different data sources, including the real Fashion-MNIST dataset and 4 GANs with different architectures. The coloured areas display how each classifier separates the input space into 10 classes; the right-most colour bar shows the colour corresponding to each class. The data points in each sub-figure are ten fixed test samples, and the number next to each point is its true label.

3.2. Methodology

To reach our goal, we create an attributor through a classifier, which involves three main stages: Surrogate Model Construction, Dataset Construction, and Attributor Construction. Additionally, we consider two attribution scenarios: Closed-World Attribution and Open-World Attribution.

Closed-World Attribution.. In this scenario, the attributor has access to all candidate GANs. Among these GANs, the attributor's objective is to determine which GAN's generated data the target model has been trained on.

- **Surrogate Model Construction.** Initially, we train multiple GAN models (i.e., candidate GANs) using diverse subsets randomly sampled from the real dataset. This process results in a collection of data sources, resulting in data generated by various GANs. For each source, we train multiple classification models, referred to as surrogate models.

- **Attributor Dataset Construction.** We use the probing set partitioned from the real dataset to query all surrogate models, assigning the same class labels to output from surrogate models trained on the same GAN. Concretely, we label the output from a surrogate model trained on CGAN_0/CGAN_1/.../WGAN_9 as 0/1/.../39, respectively. Thus, we build a multi-class dataset for training the attributor.

- **Attributor Construction** Employing classical training techniques, we train the attributor from the ground up using the multi-class dataset.

To assess the attributor, we first use these candidate GANs to generate new datasets to train multiple target models. Next, we feed the same probing set into the target models to construct an independent multi-class dataset. Finally, we evaluate the attributor's generalizability using this dataset.

Open-World Attribution. In this scenario, the attributor only accesses a subset of GANs. Even when faced with a new GAN that the attributor has never seen before, it can still determine if the given GAN generated the data that the target model has been trained on. That is, this is a more realistic and challenging scenario. Since most stages are the same as those of Closed-World Attribution, we only present the different parts here.

- **Surrogate Model Construction.** We use part of GANs to train surrogate models.
- **Attributor Dataset Construction.** We employ each accessible GAN to generate a small dataset, serving as a probing set. We query all surrogate models using each probing set, labelling the output as 1 if the GAN that generated the probing set also generated the queried model's training data; otherwise, labelling it as 0. Consequently, we build a binary dataset for training the attributor.

We use the models trained from the other GANs to assess the attributor. We query the target models using probing sets generated by each GAN and combine their output into an independent binary dataset. Finally, we evaluate the attributor's generalizability using this dataset. Note that the GANs, along with the target model trained on these GANs, have never been encountered by the attributor.

3.3. Experimental Setup

Most of the experimental setup is the same as that in Section 3.3. Here, we only highlight that in the open-world attribution, we use 80% (32) of GANs to build the attributor and reserve the remaining 20% (8) GAN models for evaluation.

3.4. Results

Closed-World Attribution. Table 3 presents the performance of our closed-world model attribution. For each of the CelebA, Fashion-MNIST, and SVHN datasets, our method has achieved the ability to distinguish classifiers trained from any GAN with an accuracy of up to 87.3%. These results demonstrate that GAN-identifying information retained in classifiers remains distinctive, regardless of variations in architectures, loss functions, or training sets among the GANs.

Open-World Attribution. As demonstrated in Table 4, our attributor achieves the highest accuracy of 97.4% for Fashion-MNIST and the lowest accuracy of 75.4% for SVHN. These results indicate our attributor's ability to link a model to the GAN generating its training data, even though our attributor has never seen these GANs and models.

4. WHY DETECTION AND ATTRIBUTION WORK

In this section, we delve deeply into the reasons behind the effectiveness of our detection and attribution techniques. Specifically, our focus is on examining the decision boundary of the model.

Table 3: The Accuracy of Closed-World Model Attribution.

Dataset	Attributor I	Attributor II	Attributor III
CelebA	0.649	0.629	**0.851**
FMNIST	0.589	0.569	**0.860**
SVHN	0.640	0.625	**0.873**

Table 4: The Accuracy of Open-World Model Attribution.

Dataset	Attributor I	Attributor II	Attributor III
CelebA	0.871	0.918	**0.955**
FMNIST	0.912	**0.974**	0.967
SVHN	0.754	0.791	**0.820**

The decision boundary is a crucial aspect of a model's behaviour. It is determined by the learned parameters and weights of the model during the training process and is essential for making predictions or assigning labels to unseen or test data. Importantly, the shape, position, and complexity of the decision boundary are directly influenced by the training data. This means that the decision boundary is intimately connected to the training data's intrinsic characteristics and is a direct reflection of the training data.

To illustrate the concept of the decision boundary, we fed a probing set to five models: one trained on real data and four others trained on CGAN/ACGAN/DCGAN/WGAN generated data, respectively. We then embedded their outputs into a 2D space using t-SNE. As depicted in Fig. 2, the decision boundary of the model trained on real data notably contrasts with those of models trained using different GANs. This observation helps explain why our detection methods can successfully distinguish between them. Furthermore, we also notice that models trained from different GANs exhibit distinct decision boundaries, which provides insights into why our attributor can successfully distinguish between them.

5. CONCLUSION

In this paper, we pioneer a systematic exploration into detecting and attributing models trained on GAN-generated data. Specifically, we construct a classifier to differentiate models trained on real versus GAN-generated data. Subsequently, we establish connections between models trained on GAN-generated data and their original source GANs. This attribution process allows GAN owners to be held accountable for any misbehaviour exhibited by these models. Our research encompasses extensive experiments and the empirical results underscore the remarkable performance of our detection and attribution methods. Furthermore, we delve deeper to reveal that models trained on different sources, e.g., real data or GANs, exhibit distinct decision boundaries and behaviours.

6. REFERENCES

[1] Karan Aggarwal, Maad M. Mijwil, Sonia, Abdel-Hameed Al-Mistarehi, Safwan Alomari, Murat Gök, Anas M. Zein Alaabdin, and Safaa H. Abdulrhman, "Has the Future Started? The Current Growth of Artificial Intelligence, Machine Learning, and Deep Learning," *Iraqi Journal for Computer Science and Mathematics*, 2023.

[2] Aayushi Bansal, Rewa Sharma, and Mamta Kathuria, "A Systematic Review on Data Scarcity Problem in Deep Learning: Solution and Applications," *ACM Computing Surveys*, 2022.

[3] Steven Euijong Whang, Yuji Roh, Hwanjun Song, and Jae-Gil Lee, "Data collection and quality challenges in deep learning: a data-centric AI perspective," *The International Journal on Very Large Data Bases*, 2023.

[4] "European union general data protection regulation (gdpr)," `https://gdpr-info.eu/`.

[5] "Data protection impact assessments (dpias)," `https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/data-protection-impact-assessments-dpias/`.

[6] "Mostlyai," `https://mostly.ai/`.

[7] "Datagen," `https://datagen.tech/`.

[8] "Ydata," `https://ydata.ai/`.

[9] "Bitext," `https://www.bitext.com/`.

[10] Luis Muñoz-González, Bjarne Pfitzner, Matteo Russo, Javier Carnerero-Cano, and Emil C. Lupu, "Poisoning Attacks with Generative Adversarial Nets," *CoRR abs/1906.07773*, 2019.

[11] Ahmed Salem, Yannick Sautter, Michael Backes, Mathias Humbert, and Yang Zhang, "BAAAN: Backdoor Attacks Against Autoencoder and GAN-Based Machine Learning Models," *CoRR abs/2010.03007*, 2020.

[12] Alexander Turner, Dimitris Tsipras, and Aleksander Madry, "Clean-Label Backdoor Attacks," in *`https://openreview.net/` (OpenReview)*, 2019.

[13] Owen Mayer and Matthew C. Stamm, "Accurate and Efficient Image Forgery Detection Using Lateral Chromatic Aberration," *IEEE Transactions on Information Forensics and Security*, 2018.

[14] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros, "CNN-Generated Images Are Surprisingly Easy to Spot... for Now," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 8695–8704, IEEE.

[15] Ning Yu, Larry S Davis, and Mario Fritz, "Attributing Fake Images to GANs: Learning and Analyzing GAN Fingerprints," in *IEEE International Conference on Computer Vision (ICCV)*. 2019, pp. 7556–7566, IEEE.

[16] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz, "Leveraging Frequency Analysis for Deep Fake Image Recognition," in *International Conference on Machine Learning (ICML)*. 2020, pp. 3247–3258, PMLR.

[17] Sharath Girish, Saksham Suri, Saketh Rambhatla, and Abhinav Shrivastava, "Towards Discovery and Attribution of Open-world GAN Generated Images," *CoRR abs/2105.04580*, 2021.

[18] "Machine learning model attribution challenge," `https://mlmac.io/`.

[19] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Deep Learning Face Attributes in the Wild," in *IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 3730–3738, IEEE.

[20] Han Xiao, Kashif Rasul, and Roland Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," *CoRR abs/1708.07747*, 2017.

[21] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng, "Reading Digits in Natural Images with Unsupervised Feature Learning," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning (DLUFL)*. 2011, p. 5, NIPS.

[22] Mehdi Mirza and Simon Osindero, "Conditional Generative Adversarial Nets," *CoRR abs/1411.1784*, 2014.

[23] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *CoRR abs/1511.06434*, 2015.

[24] Augustus Odena, Christopher Olah, and Jonathon Shlens, "Conditional Image Synthesis with Auxiliary Classifier GANs," in *International Conference on Machine Learning (ICML)*. 2017, pp. 2642–2651, PMLR.

[25] Martin Arjovsky, Soumith Chintala, and Léon Bottou, "Wasserstein Generative Adversarial Networks," in *International Conference on Machine Learning (ICML)*. 2017, pp. 214–223, PMLR.