

Utilization of GAN for Automatic Evaluation of Counterfactuals: Challenges and Opportunities

Anish Patil

School of Computing
Dublin City University
Dublin, Ireland

anish.patil4@mail.dcu.ie

Itisha Kothiyal

School of Computing
Dublin City University
Dublin, Ireland

itisha.kothiyal2@mail.dcu.ie

Vitor Horta

Insight Centre for Data Analytics
Dublin City University
Dublin, Ireland

vitor.horta@insight-centre.org

Alessandra Mileo

Insight Centre for Data Analytics
Dublin City University
Dublin, Ireland

alessandra.mileo@insight-centre.org

Abstract—Over the past few years, Explainable Artificial Intelligence (XAI) has grown significantly. This is a result of the growing use of machine learning, especially deep learning, which has produced models that are very accurate but difficult to understand and interpret. The main goal of XAI is to provide an effective approach that is simpler for humans to understand and enable better interpretability of the judgments/classifications made by the neural networks. In the XAI community, counterfactual justifications are frequently employed now because of their efficacy in explaining the mistakes, describing what updates could be done in a particular image to attain the correct classification [16].

The primary area of focus in this paper is to report on the challenges of adopting the utilization of GANs (Generative Adversarial Networks) for generating and manipulating images trained on the CelebA dataset and validation of latest research work in XAI on counterfactuals for the CUB-200-2011 [9] birds dataset.

Index Terms—Explainable Artificial Intelligence (XAI), Counterfactual explanations, Generative Adversarial Networks (GAN), Computer Vision

I. INTRODUCTION

Due to the demand for accountability and transparency when using AI models for making critical decisions, explainable artificial intelligence (XAI) has emerged as an active research area. Modern Convolutional Neural Networks (CNNs) have excelled in Computer Vision, but it is still difficult to explicit their decision-making process and validate the same, especially when errors are made.

Counterfactual explanations are a major step towards explainability and hence they are widely used in the XAI community. A counterfactual explanation, given an input question, is a version of the input with minor but significant alterations that alters the model's output conclusion [3]. For any given input query, a counterfactual explanation aims at producing a version of the input with minimal but meaningful updates that holds the capability to flip the model's resultant decision. *Minimal* refers to sparse changes that might be applied to a given image in order to produce an output image for further classification and *Meaningful* refers to the textual explanation that can be easily interpreted by a human to comprehend the information conveyed by the counterfactual [3].

A counterfactual explanation is a response to the question, "What other image, somewhat different and meaningful, might

affect the model's outcome?" for a given trained model and query image [3]. An example of a counterfactual explanation is one that is produced by altering the real data in such a way that the classifier's prediction is inverted [2]. A typical instance of counterfactual explanation generated from the model implemented in [1] is described below for the Fig. 1.

Counterfactual explanation: "If the attribute primary color of input image assumed the value brown and underparts color assumed the value brown and upperparts color assumed the value brown, input image would more likely be classified as a Black Footed Albatross instead of Blue Jay."



Fig. 1. Black Footed Albatross (ground truth) misclassified as Blue Jay (incorrect prediction). [1]

As it is clear from the above-mentioned definitions that counterfactuals are a great pathway towards explainability, however there lacks a benchmark for evaluating these semantic textual explanations. Hence, there is a strong dependency on human-intervention to validate their efficacy as a semantic explanation. Our proposal in this paper is to explore the possibility if GANs can be employed to constructively and systematically validate these semantically generated counterfactuals.

GAN (Generative Adversarial Network), a type of neural network, uses convolutions to train on images and work by playing a game of min max between a discriminator and a generator. In GANs, a discriminator is a network which trains to be able to distinguish and classify real and fake images. It returns a metric which is a probability of how sure it is that an input image is real image. A high score from the discriminator

indicates it to be a real image. The generator, on the other hand, has the task of fooling the discriminator by generating fake images. As the GAN trains, it is able to generate images that are more and more realistic to the point that they are almost indistinguishable from real images.

Our goal in this paper is to assess the quality of the above-mentioned counterfactual explanations produced by the model used in [1] by training a GAN (Generative Adversarial Network) on the CUB-200-2011 [9] birds' dataset on a specific set of attributes particularly: primaryColor, underpartsColor and upperpartsColor. We selected 15 attributes for primaryColor, underpartsColor and upperpartsColor, out of the total 312 attributes for each bird.

Our experimentation involves the process of finding a GAN that can be trained to produce perceptually good results on test samples for the CUB-200-2011 dataset on the desired semantic attributes as mentioned in the counterfactual generated in [1]. To achieve the same, we initially tried training a DCGAN on CelebA dataset, then eventually moved to StarGAN and AttGAN on the same CelebA dataset. Since CelebA dataset has 40 attributes, we implemented the respective GAN models to achieve satisfactory attribute changes through a GAN for the most commonly used CelebA dataset. We also executed the pre-trained Style-AttGAN model on CUB-200-2011 dataset. Since this model would yield nearly similar random images on the basis of the description captions hence it did not satisfy our expectations of implementing a GAN model that modifies the original image, which would help us in drawing better comparisons with respect to different approaches adopted.

The output test images from the implemented GAN models (generated on the basis of the counterfactual from [1]) were fed into the VGG-16 model pre trained over ImageNet and fine-tuned on CUB200 dataset in [1], to evaluate the classification of the test image if it gets inverted as per the counterfactual explanation. Furthermore, the resultant test images were also evaluated on metrics like LPIPS (Learned Perceptual Image Patch Similarity) [11] and FID (Fréchet Inception Distance) [10] scores. While FID takes distance of feature vectors into account and works well for diverse datasets, LPIPS helps in calculating perceptual similarity which is similar to the way a human would perceive an image.

The key challenges faced in this entire experiment include: 1) Platform, package, dependent libraries and CPU/GPU related issues for reproducing the GAN implementation on the CelebA dataset. This particularly involves the compatibility issues with respect to latest versions available for tensorflow, pytorch, python etc. 2) Low resolution images produced by the GAN and validating the same against the classifier trained on high resolution images. 3) Metric for validating the counterfactual quantitatively.

Remaining sections in the report are organised as follows: Section 2 provides information on the related works in the domain of using GANs and other methods to validate their counterfactuals. Section 3 illustrates details about the datasets used and the pre-processing tasks performed on the same before initiating the GAN model training. In Section 4, we

explain our entire experiment on the methodology adopted. Section 5 talks about the final outcomes achieved through the experiment. In Section 6, we debate over assumptions or methods that can be adopted to achieve the required outcomes, while in Section 7 we provide information on the technical challenges faced in the entire experimentation process. Finally, in Section 8, we put forth the possibilities of future work in this domain.

II. RELATED WORK

The framework described in [2], generates counterfactual visual explanations for the classifier while using a conditional GAN for transparent decision-making process in healthcare applications. The methodology revolves around perturbation of the original input image such that an explanation function is designed using cGAN keeping in mind the three properties of valid transformation namely: data consistency, classification model consistency and context-aware self-consistency. They used metrics like FID (Fréchet Inception Distance), CV (Counterfactual Validity) [12] and FOP (Foreign Object Preservation) to support the above three properties for valid transformations. FOP is a metric that they devised which helps in measuring if the patient-specific properties (foreign objects) are retained in the image like pacemaker etc. The intent of the task is similar to our experiment with a difference that in their method, counterfactuals are already images. However, in our experiment the counterfactuals are a human-readable text, that is easier for humans to interpret but difficult for testing and validating. Additionally, their work is specifically for healthcare service line and has been tested on datasets: celeba, MNIST and simulated data in comparison to ours.

The other model STEEX (STEering counterfactual EXplanations using semantics) implemented in [3] makes use of the latest advancements made in the area of semantic-to-real image synthesis in order to achieve "region-targeted counterfactual explanations" (a concept introduced by the authors), which is the highlight of the paper. Their prime attention is to spotlight the how content-based image classification is vital than only region-based classification which is absolutely true when considering scenarios where safety is of utmost importance like self-driving cars. The metrics used for evaluation in their paper include: FID, Face Verification Accuracy (FVA) and Mean Number of Attributes Changed (MNAC), where the model has been trained for CelebA, CelebAMask-HQ and BDD100k datasets. Similar to the implementation in the previous paper, even this model updates the query image to produce the counterfactual image. Their method involves no textual explanations for producing a counterfactual image which is relatively easier to construe by a human for better understanding as used in our model.

Another methodology discussed in [4] talks about Causal Concept Effect (CaCE) measure for explainability and to get rid of the errors arising from confounding. Their model is based on Variational AutoEncoder (VAE), to estimate VAE-CaCE metric, which as proposed in the paper can estimate the true concept causal effect. In order to showcase the

effectiveness of the CaCE metric, they have tested their model on different datasets like MNIST, COCO Miniplaces and CelebA which clearly exhibits the generalizability of their implemented framework. But however, no GitHub repository links have been shared to support the reproducibility of the code for further research. This paper implements StarGAN for producing the counterfactual images, which is similar to our experiment where we have also used StarGAN to produce counterfactual images for celebA and CUB-200-2011 datasets. However, their experiment does not mention about semantic explanations used for producing counterfactual images but is limited to the causal concept effect.

PIECE (Plausible Exceptionality-based Contrastive Explanations) algorithm is one more illustration, which combines a GAN that creates counterfactual and semi-factual explanatory images with a CNN that makes predictions that need to be explained on datasets [13]. This model PIECE uses semantic explanations as a base to generate their counterfactual images just like our model. However, PIECE model has been tested only for datasets: CIFAR-10 and MNIST. Furthermore, Figure 7 from [13] showcases the generated image of a counterfactual explanation for a bird using PIECE and Min-Edit as an image generation model for comparison. It is easily visible to the human eye that the images have a very low resolution, and additionally there is no mention of the resolution of images employed for training the classifier and the counterfactual image generating GAN. Considering the results showcased in the paper, their model PIECE performs better in comparison to other methods like Min-Edit, C-Min-Edit, Proto-CF (Interpretable Counterfactual Explanations Guided by Prototypes) and CEM (Contrastive Explanations Method) for the five metrics presented in Table 1 of the report and scores low only for the optimization time taken for each image, which as per the authors can be compensated by utilizing a GPU or reducing the number of epochs.

As it is clear from the above-mentioned models that research in this specific domain of adding explainability to counterfactuals is either limited to a specific domain like healthcare or have been tested on simpler datasets like CelebA, MNIST, CIFAR etc. Therefore, we had to take a step further while implementing our model by parsing the complex dataset of CUB-200-2011, which contains 312 attributes for each bird with a total collection of 11,788 images belonging to 200 class labels. Additionally, we needed latest metrics which could measure the feature vector distance in the images and also indicate their perceptual similarity like FID and LPIPS scores.

III. DATASET AND FEATURES:

We chose to use the following open-sourced datasets: **CelebA** dataset [14] collected by the researchers at the Chinese University of Hong Kong having the purpose of face attributes recognition as well as face detection. The large-scale dataset consists of more than 200,000 face images of various celebrities having a great diversity of attributes and poses. Each image in the dataset has 178x218 pixel RGB dimensions and is annotated with 40 different binary attributes and 5 major

landmark locations. CelebA dataset has rich set of annotations which ensures that the machine learning projects are scalable as it is a human led task of identifying and labeling specific data in the images, in this case the attributes of celebrity faces in the dataset, and makes it easy for networks like GANs to identify and classify information like humans do and to change the annotated attributes of the images which is one of our goal in this paper.

Our main motivation for using the CelebA dataset was to validate the feasibility of using GANs to generate modified images based on the attribute changes and not to actually validate counterfactual explanations generated by [1]. The dataset was also used to check how well the GANs generated and manipulated the attributes of the images and based on the results to choose the best out of those GANs which would act as a base pipeline for further working on CUB-200-2011 dataset. It is useful to note that the CelebA dataset suffers from biases as there exists a greater majority of female celebrity faces as compared to male celebrity faces and thus the GANs are prone to generating an image with face having more feminine attributes. To add to this, the face images in the dataset do not have frontal-angle and hence are more difficult to train to capture the semantic attributes of the images.

CUB-200-2011 dataset which is an extended version of CUB-200 dataset and has 11,788 images from 200 different birds. Each image in the dataset has annotations: 15 part locations, 312 binary attributes and 1 bounding box. We randomly select 2,000 images as a test set and use all remaining images for training data for StarGAN. The main challenge of this dataset is that there is a vast variation and confounding features in background information compared to subtle inter-class differences in birds of this dataset. Our primary focus for using this dataset in our experiments is to achieve both the things : Generate and modify images by changing the attributes on the base pipeline of GAN and; Validate the counterfactuals generated by [1].

A. Data Preprocessing:

For the CelebA dataset, the images were already appropriately aligned and cropped. Further cropping of the images to the dimensions 64x64 and resizing was done to reduce the size and background clutter. As was done in the original implementation of the DCGAN, for training images and scaling them to the range of $[-1, 1]$ of the tanh activation.

For the CUB-200-2011 dataset, each image is of different dimensions and resolution hence making it a complex dataset for training purposes of GAN as there is a difficulty in training larger images. In case of GANs, the maximum resolution of generated images is limited to the resolution of real images that are used as the training set [14]. Also, the images in the dataset are not aligned and cropped. An image size of 224x224 dimensions was taken so as to get maximum resolution for the training purpose and a crop size of 512x512 was taken so that each image in the dataset is clearly visible to the generator and discriminator networks in the GAN, hence increasing the overall accuracy of generated images.

At the moment, there exists no consolidated file for all the binary attributes of the images in the CUB-200-2011 dataset. Hence, to continue our research we developed a data-parser in python to create an attributes file which indicates the presence and absence of all the 312 binary attributes for the CUB-200-2011 dataset. The presence of the attribute is indicated by ‘1’ and the absence is indicated by ‘-1’ in the final attributesAnnotations.txt file. To fetch the required attribute IDs, Image IDs and corresponding Class labels for creating the training labels we utilized: attributes.txt, images.txt and imageAttributeLabels.txt files from CUB-200-2011 dataset and parsed the information as per the required format to initiate the training on the selected attributes as provided in Table 1 and Table 2. Most of the experiments featuring GANs are performed on the famous CelebA dataset and hence, the parsed annotations file created by us paves a way for reproducing the results of such GANs for changing the semantic attributes of birds in the dataset CUB-200-2011.

B. Training

For DCGAN, the model was trained with the L2 loss and the Adam optimizer with learning rate of 0.0002. It consists of 9 convolutional layers which were separated by batch normalization and Leaky ReLU and followed by one fully connected layer. Training takes about a day as it was done only using the CPU. For DCGAN, while training the autoencoders, the images in the dataset were split into a 70/15/15 ratio for training, testing and hyper-parameter tuning respectively.

In case of StarGAN, the model are trained using Adam optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The images were flipped horizontally with a probability of 0.5 for the purposes of data augmentation. One generator update after five discriminator updates was performed. The batch size was set to 10 for all the experiments. Learning rate of 0.0001 was taken for the first 100,000 iterations and linearly decay the learning rate to 0 over the next 100,000 iterations of training. Training takes about 13 hours on a single Nvidia RTX3060 GPU.

TABLE I
ATTRIBUTES FOR TRAINING STARGAN ON CUB-200-2011 DATASET

Attributes
has_primary_color_brown
has_underparts_color_brown
has_upperparts_color_brown
has_primary_color_black
has_underparts_color_black

We reproduced the tensorflow implementation of AttGAN [7] for CelebA dataset initially and thereafter on the custom cartoon dataset. The model uses 128x128 images for training and is trained on the Adam Optimizer ($\beta_1 = 0.5$, $\beta_2 = 0.999$) with $2e-4$ learning rate. The model was trained for 60 epochs on a batch size of 32 for CelebA dataset, 500 epochs on a batch size of 32 for cartoon dataset and 20 epochs on a batch size of 35 for CUB-200-2011 dataset.

TABLE II
ATTRIBUTES FOR TRAINING ATT-GAN ON CUB-200-2011 DATASET

Attributes		
Primary color	Underparts color	Upperparts color
Blue	White	White
Brown	Grey	Black
Black	Blue	Grey
Grey	Black	Blue
White	Brown	Brown

IV. METHODOLOGY

Our approach focuses on generating and modifying the attributes of the images by using GANs and after a base pipeline is created to achieve the same, is to check if GANs could be used to assess and validate the counterfactual explanations generated by [1] by changing the attributes of the original image belonging to the CUB-200-2011 dataset that could invert a wrong prediction from wrong class to that of the correct class.

We try to implement DCGAN-encoder network [5], AttGAN [7], pre-trained Style-AttnGAN [8] and StarGAN [6] for the purpose of achieving our objectives in this paper. DCGAN is a deep convolutional GAN with encoder networks for the generator and works by taking an input image and generating a similar image with specific characteristics by manipulating the vectors, given by z vectors, in the latent space [5]. StarGAN is an image to image translation model for multiple domains [6] which can change only one attribute of the image in this experiment whereas, AttGAN aims on changing more than one attributes of a single image at a single time [7]. On the other hand, a pre-trained Style-AttnGAN [8] is a model which takes text captions as an input and generate images based on those captions.

For creating a pipeline, our experiments focused on using CelebA dataset, to instill the capability of generating an image by GAN which closely resembles the original image and manipulating the attributes from the total 40 binary attributes like ‘Pale Skin’, ‘Smiling’, ‘BlondHair’, ‘Bangs’, ‘Bald’, etc. of the original image and getting a resultant generated image with the changed attributes.

A. DCGAN-Encoder networks

We took our first step with the DCGAN implementation as depicted in [5] for the CelebA dataset to see if we could generate an image using DCGAN and use the encoder-decoder approach on top of the trained DCGAN for manipulating the attributes of the images in the dataset.

Generative model was trained first by using deep convolutional generative adversarial networks. The model was trained for 25 epochs for the entire CelebA dataset, and learning rate of 0.0002 was used by using the Adam optimizer until the loss curve converged [5].

The trained generator and discriminator of the DCGAN was then used to build the encoder architecture. The GAN auto-encoder is used in such a way that it takes an input image and

produces a z vector in the latent space, which produced an image resembling close to that of the original image. CelebA dataset was divided into training, validation and test sets. To train the encoder model, training set of the CelebA was used by giving the input of an image from the dataset and measuring the loss as the result of any given similarity metric. Finally, we sampled the images from the validation set and run them through the encoder and decoder architecture to get the qualitative results.

The attributes in the CelebA dataset were represented as vectors in the latent space and each image in the dataset is labeled with one of the 40 binary attributes where 1 denotes the attribute is present and -1 denotes the attribute being absent for that corresponding image. Z vectors representing these attributes were calculated first by subtracting the average z vectors of all the images which do not have the specific attributes from the average z vectors of all the images which have the specific attributes in the training data [5]. The images were manipulated by first encoding the image in the latent space by using the encoder network and then adding the attribute z vector to the encoded image vector. This was done for manipulating the images with different attributes.

B. AttGAN and Style-AttnGAN

Next steps of the experiment comprised of utilising AttGAN [7] for CelebA dataset to check the clarity of the results during sampling. As can be seen in Fig. 2, the AttGAN networks provided satisfactory results on the CelebA dataset using the training parameters mentioned before for changing the facial attributes of the images in the dataset.

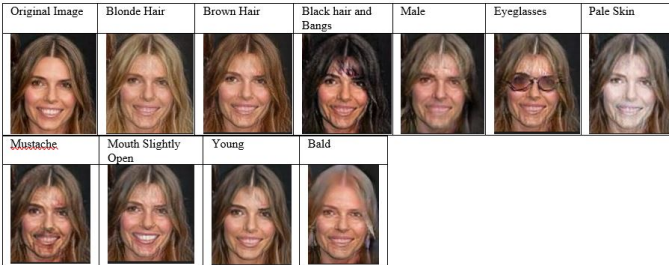


Fig. 2. AttGAN [7] Implementation results for CelebA dataset for a model trained on 13 attributes



Fig. 3. AttGAN [7] Implementation results for custom cartoon dataset for 20 attributes

We further checked the AttGAN implementation for another custom dataset i.e., cartoon dataset (which has 18 attributes and 10,000 cartoon face images). The results for both training and testing samples are presented in the Fig. 3. Next step

for using AttGAN was to make it work for the CUB-200-2011 dataset to get similar or better results in comparison to StarGAN for changing the semantic attributes of the birds. However, as it is evident from Fig. 6, which highlights the training sample at epoch 19, the networks failed at manipulating the attributes of the images and generated same image for all the image samples. Even though the images generated by AttGAN were close to the real images, it could not generate the images with different attributes. One possible reason for this could be due to issues in parsing the annotations of the attributes for the images in the CUB-200-2011 dataset and hence, the AttGAN network could not detect and thereby, manipulate the attributes of the images.

Simultaneously, we also executed the pre-trained model of Style-AttnGAN [8] on CUB-200-2011 dataset, which transforms the text captions provided to the model into images. Since our goal through this experiment is to validate the textual counterfactual explanations, so this approach of Style-AttnGAN [8] would have proved advantageous if it could produce attribute changes as mentioned in the text captions to the same image of which the counterfactual explanation was produced. But Style-AttnGAN [8], produces random images of the birds based on the text captions provided for test-samples. The resultant bird images fetched post testing are described in Fig. 4 and Fig. 5, along with their variation from the reference image. The text captions provided here were based on the counterfactual explanation generated from the VGG-16 [1].

As it is clear from the Fig. 4 and Fig. 5, the mentioned captions generated random images of the birds which could not satisfy the preliminary requirement for the validation of the counterfactual as illustrated in the example provided in the Introductory section of the paper.

Reference Bird Name	Counterfactual Explanation	Converted Caption for Style-AttnGAN	Original Image from CUB-200 Dataset	Generated Image from Style-AttnGAN
Black Footed Albatross	If the attribute primary color of input img001 assumed the value brown and underparts color assumed the value brown and upperparts color assumed the value brown, input img001 would more likely be classified as a Black Footed Albatross instead of Blue Jay	bird with primary color as brown and underparts color as brown and upperparts color as brown		
Black Tern	If the attribute underparts color of input img002 assumed the value black and primary color assumed the value black, input img002 would more likely be classified as a Black Tern instead of Elegant Tern	tern bird with underparts color as black and primary color as black		

Fig. 4. Images generated from Style-AttnGAN [8] from the text captions.

C. StarGAN

Our next approach was to check and utilise StarGAN [6] implementations for CelebA dataset to check the feasibility,

Bird Name	Original Image	Image generated	Image Caption Reference	LPIPS score (AlexNet)	FID
American Crow			American_Crow_0031_25433	0.4694	226.78
Yellow billed Cuckoo			Yellow_Billed_Cuckoo_0008_26578	0.6598	233.39
Purple Finch			Purple_Finch_0011_27633	0.5880	173.20
Indigo Bunting			Indigo_Bunting_0034_12464	0.5042	125.33
Gray crowned rosy finch			Gray_Crowned_Rosy_Finch_0022_27028	0.5663	194.07

Fig. 5. FID and LPIPS scores for reference image and resultant image from Style-AttnGAN [8].

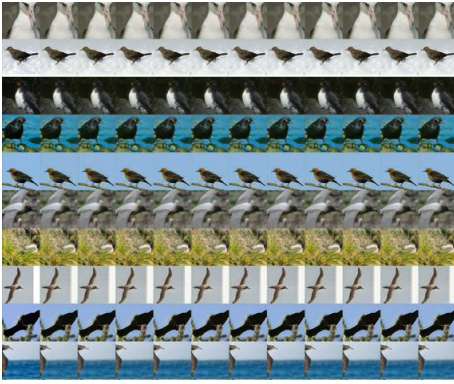


Fig. 6. Training sample at epoch 19 iteration 215 for AttGAN Implementation on CUB-200-2011 dataset.

code reproducibility and the clarity of the results obtained. StarGAN, proposed in [6] is a scalable image-to-image translation model meaning it aims on changing a particular aspect of a given image to another image by using a single generator and a discriminator of the generative adversarial networks. Fig. 8 shows the facial attribute transfer results on CelebA dataset for the facial attributes 'Black Hair', 'Blond Hair', 'Brown Hair', 'Male' and 'Young'. The method proposed in [6] provides considerable higher visual quality on test data compared to the experiment using DCGAN-encoder networks on CelebA dataset. This could be due to regularisation effect of StarGAN using a multi-task learning framework [6]. An interesting approach followed by using StarGAN is to train the model to flexibly translate images according to the labels of the target domain rather than training the same model to perform a static and fixed translation which could lead to overfitting of the model.

StarGAN when compared with DCGAN-encoder network, AttGAN and Style-AttnGAN provided better results for achieving one of our primary task in this paper, to validate the feasibility of using GANs to generate modified images based on the attribute changes. The images thus obtained using CelebA dataset assures that GANs in fact are a viable method to generate and manipulate the images and produces an images closely resembling the original image by tweaking some aspects of the original image.

D. Validating the counterfactual explanations using StarGAN

The next primary goal of this paper is check if the GAN pipeline that was created to generate and manipulate the images based on some attributes, could be used to assess and validate the counterfactual explanations generated by [1] by changing the attributes of the original image belonging to the CUB-200-2011 dataset that could invert a wrong prediction from wrong class to that of the correct class.

Out of all the GANs we implemented for generating and manipulating the images of the CelebA dataset, StarGAN performed as expected and resulted in generating a visually appealing image. Hence, StarGAN was chosen as the candidate among other GANs for assessing the counterfactual explanations in [1]. Our next steps to validate the counterfactual explanations generated in [1], involved replicating StarGAN for the CUB-200-2011 dataset. At present, there exists no GAN implementation that parses the complex dataset of CUB-200-2011 for training the model on a set of attributes selected from the total 312 binary attributes of the CUB-200-2011 dataset.

Hence, to make this experiment work, we developed a data parser to create an annotations file which indicates the presence and absence of the selected attributes for training the model on CUB-200-2011 dataset. The key idea of making this data parser is to create a file, similar in the format of the binary attributes for the images in CelebA, but for the all the images in the CUB-200-2011 dataset. The images in the CUB-200-2011 dataset were consolidated in a single working directory with the annotations files created by us alongside. StarGAN was trained using the same parameters for the CUB-200-2011 dataset as was for CelebA dataset.

Initially, we used an image size of 128x128 of the images to train the model and selected the attributes mentioned in table I. StarGAN was trained only on these 5 semantic attributes out of 312 of the CUB-200-2011 dataset as our primary focus was evaluate the counterfactual explanations in [1] by using the image-to-image translation model of StarGAN for these selected attributes. Later, an image size of 224x224 of the training images was used to yield a better resolution of the generated image from StarGAN. A crop size of 512x512 was taken while training the model as the images in the CUB-200-2011 dataset are not aligned and cropped properly and hence for the GAN networks to generate an image which resembles the real image, the image had to be visible by taking the crop size.

V. RESULTS

Results can be viewed by checking our two main objectives of this paper: 1. Qualitative results of creating a pipeline using the CelebA dataset to generate an image by GAN closely resembling the original image and also for changing the attributes of the images to get a generated image are given for the following GANs:

A. DCGAN-Encoder result

The qualitative results obtained from the trained DCGAN-Encoder model which were generated by first generating z encoding vectors and later decoded by the generator for obtaining the images. The generated images were not satisfactory as the images obtained from the sample-training and testing dataset were not clear enough and suffered from blur and generated an image with more feminine attributes due to bias in the dataset to detect the face attribute changes as depicted in the Fig. 7. Upon using the Encoder networks on top of the trained discriminator, to generate an image with changed attributes were not satisfactory as well. This could be possibly due to the blur and non-realistic generated image from DCGAN and getting an image with the desired attributes by using the same image. This could be seen evidently from Fig. 7






Original Image	DCGAN-Encoder generated image	Bald	Attractive	Pale Skin
				

Fig. 7. DCGAN [5] Implementation results for CelebA dataset for a model trained on 40 attributes

B. StarGAN results





































Original Image	Black Hair	Blond Hair	Brown Hair	Male	Young
					
					
					
					
					
					

Fig. 8. StarGAN [6] Implementation results for CelebA dataset for a model trained on 5 attributes

The qualitative results obtained for StarGAN on CelebA dataset using the training parameters mentioned earlier were satisfactory as depicted in the Fig. 8. StarGAN implementation resulted in generating better images by manipulating the

attributes of the images in the dataset. The images, as can be seen, are less blurry and can distinctively generate images with different attributes for an image size 128x128.

The qualitative results obtained for StarGAN on the CUB-200-2011 dataset were satisfactory enough visually and resulted in better quality of images by changing the semantic attributes for the dimensions 128x128 and 224x224 as can be seen in Fig. 9






















Dimensions	Original Image	Primary color brown	Primary color black	Upper parts color brown	Under parts color black	Upper parts color brown
128x128						
128x128						
224x224						
224x224						

Fig. 9. StarGAN [6] implementation results for CUB-200-2011 dataset for a model trained on 5 attributes in two different dimensions

Quantitative results using LPIPS and FID score: The lower values of LPIPS and FID as represented in Fig. 10 indicates better perceptual similarity of the images generated by the StarGAN by manipulating the attributes of primary color black and primary color brown for both Black Footed Albatross and Black Tern. It is interesting to note here that LPIPS and FID values for the both the images for the attributes given in Fig. 10 was lower for the resolution 224x224 than for resolution 128x128 meaning better perceptual similarity for the images generated with higher resolution.







Bird Class	Original Image	Generated Image	Attribute Change	LPIPS	FID
Black Footed Albatross			has_primary_color_black	<ul style="list-style-type: none"> 0.1943 (128*128) 0.1210 (224*224) 	<ul style="list-style-type: none"> 321.82 (128*128) 153.00 (224*224)
			has_primary_color_brown	<ul style="list-style-type: none"> 0.1936 (128*128) 0.1242 (224*224) 	
Black Tern			has_primary_color_black	<ul style="list-style-type: none"> 0.2573 (128*128) 0.2122 (224*224) 	<ul style="list-style-type: none"> 322.43 (128*128) 290.90 (224*224)
			has_primary_color_brown	<ul style="list-style-type: none"> 0.2972 (128*128) 0.2318 (224*224) 	

Fig. 10. Metric values for generated images using StarGAN on CUB-200-2011 dataset with 128 x 128 and 224 x 224-pixel resolutions.

2. Results for checking if StarGAN could be used to assess and validate the counterfactual explanations generated by [1] by inverting a wrong prediction from wrong class to that of the correct class can be seen in Fig. 11. The resultant higher resolution (224x224) images were then provided as inputs to the VGG-16 model [1] to check if the wrong classification for Black Footed Albatross is flipped.









Original Image	Attribute selected	Generated Image	Classification
Black Footed Albatross	has_primary_color_brown		Horned_Lark
	has_underparts_color_black		Spotted_Catbird
	has_underparts_color_brown		Tennessee_Warbler
	has_upperparts_color_brown		Horned_Lark
Black Tern	has_primary_color_black		Scissor_tailed_Flycatcher
	has_primary_color_brown		Orange_crowned_Warbler
	has_underparts_color_black		Spotted_Catbird
	has_upperparts_color_brown		Orange_crowned_Warbler

Fig. 11. Classifier Decision for the StarGAN generated images at 224 x 224 dimensions.

The images generated by the StarGAN are of the resolution size: 128x128 pixels and the major drawback with the CUB-200-2011 dataset is that the images have different sizes, hence for the first training model on StarGAN we used image size = 128 and crop size = 512 pixels, which did not yield to good resolution images [14] for the classifier in [1] to present accurate results. Hence, we trained StarGAN model again to get the image size = 224x224 pixels. Even at this resolution, the classifier [1] experiences a lot of noise and hence throws random classification outcomes as presented in Fig. 11

VI. DISCUSSION

The images generated by altering each of the semantic attributes given in Table 1 using GANs in accordance with the counterfactual explanations improves the probability of the expected class in some cases. Possible reason why the prediction score is not as expected even after changing the semantic attributes is due to the lower resolution of the generated images. Moreover, attributes for which we obtained better probabilities could be ranked as the top predictors of getting the expected class in some cases.

Based on our experiments on AttGAN, StarGAN and DC-GAN, we realized that the output images obtained are of relatively lower resolution; possessing a lot of noise when it comes to validating classifier's predictions. Hence, it appears

to be a limitation while producing images with semantic attribute updates with the above-mentioned GANs.

The images generated from the StarGAN could not invert the classification as it could be argued that the resolution of the generated image plays a vital role in the classifier's prediction. To add to this, the prediction score of the expected class improves for the image of a Black footed albatross used in [1] generated by StarGAN for the attributes '*PrimaryColor-Brown*' and '*UpperpartsColor-Brown*' of resolution 224x224 when compared to the same image generated by StarGAN for same attributes at 128x128 resolution. However, this was not the case for the attribute '*UnderpartsColor-Brown*'.

In case of generated image of a Black Tern used in [1] by StarGAN, the prediction score of the expected class improves for the attribute '*PrimaryColor-Black*' of resolution 224x224 when compared to the same image generated by StarGAN for the same attribute at 128x128 resolution. However, this was not the case for the attribute '*UnderpartsColor-Black*'

Considering the aforementioned results, we can assume that if the image classifier was trained on lower resolution images of the CUB-200-2011 dataset (128X128 or 224X224) we could obtain better probabilities of the expected classes and could arguably invert the classification based on the counterfactual explanation. It can also interesting to argue that StarGAN trained on CUB-200-2011 dataset, does not recognise the attribute '*UnderpartsColor*' very well and hence the prediction score of the expected class does not improve even on increasing the resolution of the generated image by StarGAN.

VII. TECHNICAL CHALLENGES

Training of these GAN models on CPU takes extensively long time as we increase the number of epochs. Hence use of a GPU is an advantage when considering to work with training of GANs, to obtain better results and validate the same simultaneously, while not worrying about time. While implementing GAN models on tensorflow, we realized that most of the models employ the usage of tensorflow 1.15 which is compatible with Python version 3.6 and lower. If one uses higher versions of tensorflow and Python, there occurs many import issues with respect to various packages like trace, absl-py etc that requires code changes in the training python files and extra debugging. Additionally, working with the tensorflow implementation of AttGAN for version 1.15, by default it installs tensorflow-estimator version as 2.0 or higher which needs to be taken care of separately by installing a lower version of 1.15 for tensorflow-estimator, which would otherwise lead to code errors.

VIII. FUTURE WORK

In further iterations of the StarGAN, the model can be trained on more attributes and outcomes obtained can be used to validate more counterfactuals which would prove advantageous in calculating Counterfactual Validity (CV) [12] scores,

since there would be enough counterfactual validation results to do so. CV as an additional metric would greater evidence to the counterfactual evaluation for model in [1]. Another step in the validation process would be to utilize the link prediction scores from [1] to produce different counterfactual images from the trained GAN and thereby validate the link prediction approach used in [1].

IX. CONCLUSION

Through the experiments performed, we aimed at building and checking a wide range of GANs for the purpose of image generation which resembles the original image and which possess the capability of allowing the update of specific attributes on the CUB-200-2011 dataset. Our primary goal was also to use such generated GAN image by changing the semantic attributes of the original image to access and validate the textual counterfactual explanations in [1]. Since, there exists no current research work in this domain, we initiated with CelebA dataset to create a baseline pipeline for generating and modifying the images based on their attributes. Post creating a data-parser for the CUB-200-2011 dataset, we trained GAN models to obtain resultant images with the particular attributes specified and to check if the GANs in general could be used as a way to systematically validate the counterfactual explanations generated in [1]. The whole experimental process was focused at achieving good resolution generated images with semantic attribute changes as per the counterfactual explanations obtained from [1] so as to assess the same by feeding the outputs back to the VGG-16 model in [1]. StarGAN as described in the Fig. 8 and Fig. 9, produced much better results than DCGAN and hence added value to the counterfactual images produced, which proved quite valuable in the whole validation process. Since there exists not much research in the area of highlighting the limitations for validating the semantic counterfactual explanations by using GANs, our experiments give a brief overview of the limitations faced by GAN in this particular field and additionally explores the untouched complex CUB-200-2011 dataset with respect to updating semantic attributes.

REFERENCES

- [1] Horta, V., Mileo, A.: Generating textual explanations for CNNs using knowledge graphs. In: AIXIA 2021 - Advances in Artificial Intelligence - XXth International Conference of the Italian Association for Artificial Intelligence, December 1-3 (2021). In Press. (Accessed: 12 November, 2021).
- [2] Singla, S., Pollack, B., Wallace, S. and Batmanghelich, K. (2021) 'Explaining the Black-box Smoothly- A Counterfactual Approach, Available at: <https://arxiv.org/pdf/2101.04230.pdf> (Accessed: 24 January 2022)
- [3] Jacob, P., Zablocki, E., Ben-Younes, H., Chen, M., Perez, P., Cord, M., Valeo.ai and Universite, S. (2021) 'STEEX: Steering Counterfactual Explanations with Semantics', Available at: <https://arxiv.org/pdf/2111.09094.pdf> (Accessed: 25 January 2022)
- [4] Goyal, Y., Feder, A., Shalit, U. and Kim, B. (2020) 'Explaining Classifiers with Causal Concept Effect (CaCE)', Available at: <https://arxiv.org/pdf/1907.07165.pdf> (Accessed: 22 January 2022)
- [5] R. Homero, B. Roman, M. Yang, and Zhang, "Photoshop 2.0: Generative Adversarial Networks for Photo Editing." Accessed: Aug. 08, 2022. [Online]. Available: <http://cs231n.stanford.edu/reports/2017/pdfs/305.pdf>

- [6] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation," arxiv.org, Nov. 2017, [Online]. Available: <https://arxiv.org/abs/1711.09020>
- [7] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "AttGAN: Facial Attribute Editing by Only Changing What You Want," IEEE Transactions on Image Processing, vol. 28, no. 11, pp. 5464–5478, Nov. 2019, doi: 10.1109/tip.2019.2916751.
- [8] T. Xu et al., "AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks," arXiv.org, 2017. <https://arxiv.org/abs/1711.10485>
- [9] "Perona Lab - CUB-200-2011," www.vision.caltech.edu/datasets/
- [10] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium," arXiv:1706.08500 [cs, stat], Jan. 2018, [Online]. Available: <https://arxiv.org/abs/1706.08500>
- [11] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric," arXiv:1801.03924 [cs], Apr. 2018, [Online]. Available: <https://arxiv.org/abs/1801.03924>
- [12] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, Jan. 2020, doi: 10.1145/3351095.3372850.
- [13] E. Kenny and M. Keane, "On Generating Plausible Counterfactual and Semi-Factual Explanations for Deep Learning." Accessed: Aug. 05, 2022. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/17377/17184>
- [14] Park, M.; Lee, M.; Yu, S. HRGAN: A Generative Adversarial Network Producing Higher-Resolution Images than Training Sets. Sensors 2022, 22, 1435. <https://doi.org/10.3390/s22041435>
- [15] "Image Editing using GAN," Image-Editing-using-GAN. <https://tandon-a.github.io/Image-Editing-using-GAN/> (Accessed 05 January, 2022).
- [16] R. M. J. Byrne, "Counterfactuals in Explainable Artificial Intelligence (XAI): Evidence from Human Reasoning," www.ijcai.org, pp. 6276–6282, 2019, [Online]. Available: <https://www.ijcai.org/proceedings/2019/876>