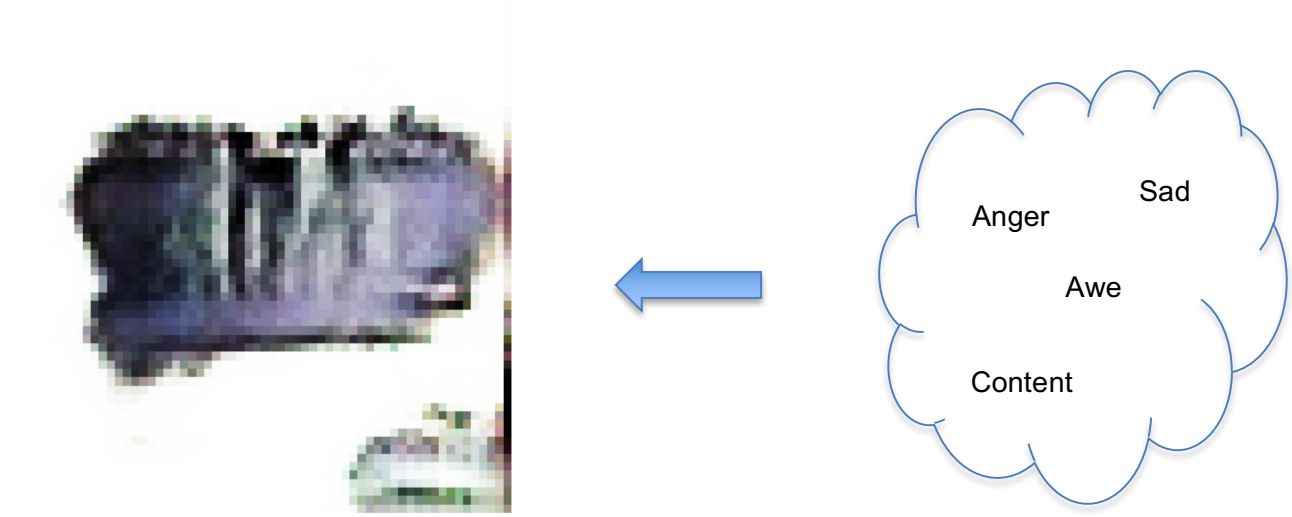
Machine Learning for the Arts
UCSD FALL 2019
FINAL PROJECT

Networks Got Talent! - Network Judging Other Network's Album Art



Shawheen Tosifian

The goal of this project was to see, more as a philosophical exercise, the outcome of a neural network assigning an emotion (label essentially) to an image (an 'album cover') created by another network. Ideally, given that neural networks are still a collection of quantitative methods, it's an experiment in seeing how emotion and feeling can be seen reduced to a controlled dialogue between a generator and classifier dealing with 64 x 64 images (hardly a romantic viewpoint, of course). This project is an extension of the generative visual project explored earlier where a network was trained on unlabeled album covers and generated its own. The outputs of that project were taken and fed into a classifier trained on abstract art images labeled with different sentiments (i.e. 'anger', 'contentment'). The results were variable as predicted but that can be attributed to the nature of the experiment.

Concept

One domain that we often use to separate machines from humans is that of emotion. Emotion seems to serve a biological purpose in living beings, however for a machine, it can be seen as the equivalent of an appendix. This is mostly why pop cultural depictions of robots/A.I.s are often as cold, calculating automatons with no regards for morality and emotions associated with it. Of course, this project does not get anywhere near these implications or shine a new light on them, but just a playful take on the idea of A.I.s (in our case, crude neural networks) carrying out tasks we associate as very human (creating art and judging its response).

The reason for choosing to generate album artwork lies in the previous project. Originally, the goal was to train a GAN on album artwork from every decade (50s and onwards) with the intention that, when properly trained, the GAN would produce artwork evocative of the visual style of the decade. This idea never came to fruition due to issues with scraping album artwork labeled by decade from the iTunes and Spotify APIs. As a result, the GAN was trained on unlabeled album artwork from all decades and created artwork based off these samples. The reason for choosing to label the sentiment of these generated album covers for this project lies mostly in the idea of what the purpose of an album cover is. Is it used to convey the emotion of the music inside it? Or is it purely for visual appeal? Or something else entirely? All of those propositions are valid due to the wide uses of album artwork throughout popular culture. The goal for this project goes back to the idea of seeing if emotion can be 'reduced' to a quantitative approach and treating album covers as artwork (generated ones at least) that can be used as something conveys emotion.

Technique

Generative Adversarial Networks (GANs)

A generative adversarial network is two networks (a generator and discriminator) in a 'game' with each other, trying to outsmart the other (a common analogy is that of a counterfeiter and detective). This process works in terms of data distributions and latent spaces. The generator learns to map from a latent space to a data distribution of the one it's trying to replicate (so in this case, album artwork). Meanwhile, the discriminator is learning to differentiate between the fake samples and the true data distribution. As this 'game' continues, both networks ideally become better at replicating and differentiating, resulting in even better fake samples and

better differentiation of them. Both networks use backpropagation to update their weights and improve their performance. The GAN used for this project is a deep convolutional GAN, using convolutional layers to get better performance.

VGG16 Classifier

For classifying the emotion/sentiment of each GAN album art sample, an classifier based on the VGG16 model (pre-trained for image recognition) was to assign an emotion label for each image. VGG16 is also a deep convolutional neural network, using many 3x3 kernel-sized filters after another.

Process

Initially, for the album cover generation portion of the project, a data scraper tool called CoverPy was used to grab album covers from the iTunes API. The tool however would have buffering issues and would fail in the middle of its run. Moreover, it wasn't grabbing enough data for the GAN to train (only in the hundreds) due to the way the data was fed into the API. Therefore the idea to train on albums by decade was ultimately tossed. It was then decided to train the GAN an unlabeled album cover dataset found on archive.org (a relatively small one of 5k+ images). The DCGAN was then trained on this set for roughly 70 epochs and after, 32 generated samples were saved. Next, the VGG16 classifier was trained on a labeled abstract art dataset found on imageemotion.org.

(a) (b)

Figure 1: (a) Some of the album covers used to train the DCGAN (b) Two images from the abstract art dataset used to train the VGG16 classifier

For all training and classification, images of size 64x64 were used for simplicity of the model as well as shorter training and classification time. They were resized prior to training using basic image processing libraries in Python (PIL etc.). The labels for the abstract art were reformatted for classification since the original labelling assigned a value for each emotion the artwork, e.g. a piece might have a 'Anger' value of 8 (from 0-10), 'Fear' value of 5, and so on. This was reduced to one emotion label by taking the emotion label with the highest value and assigning the piece with that emotion, reducing the problem to that of classification (regression was difficult to implement here, though is worth a closer look for a future implementation). The network was then trained using these labels and then the GAN generated album covers were passed to it for classification.

Result

The GAN generated images were interesting in that you could make out some instances where it looked like it was trying to replicate a line up of faces that are often found on album covers, but still underdeveloped. Everything else came out sort of abstract looking coincidentally (see below), with a variable but subdued palette of colors and a lot of repeating patterns (could be attributed to smaller than average dataset for a GAN). Passing these images into the classifier resulted in somewhat agreeable results.

Figure 2: (a) Some of the GAN album covers generated and passed to the classifier
(b) Labels given to the GAN album covers by the classifier

Reflection

I thought the classifications given to these samples were interesting and could definitely work. The interesting thing with this idea is that if the classifier was trained differently/initialized with different weights etc., it would output different labels for the same piece. I find this interesting because the same can be said of human interpretation of art, which is highly associative. Someone might have an entirely different reaction to a work of art than someone else due to their own 'training' or experiences in life (basically having a cognitive bias toward a piece), which is a neat parallel to a neural network's training (i.e. the dataset it was trained on == it's 'life experiences'). Again, it's very easy to look too much into these results given the crudeness in which they were attained, but I believe this could be extrapolated to much larger and robust architectures trained on much more images (and higher resolution ones).

CODE: https://github.com/ucsd-ml-arts/ml-art-final-shawheen-t-final