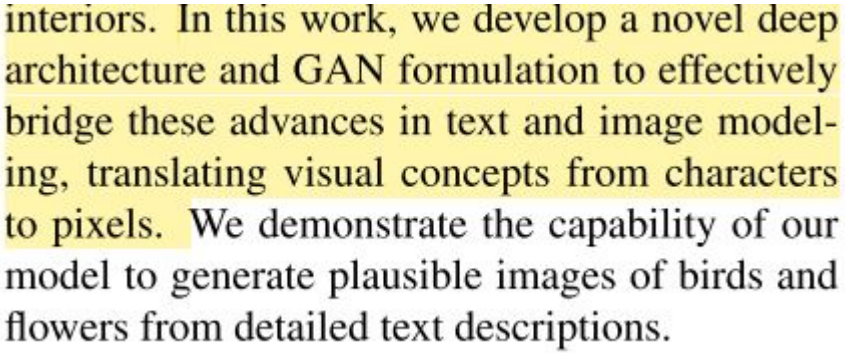
Generative Adversarial Text to Image Synthesis

-Nikunj Gupta
-AI Reading Group

Abstract

- Automatic synthesis of realistic images from text would be interesting and useful, but current AI systems are still far from this goal.

- Meanwhile, deep convolutional generative adversarial networks (GANs) have begun to generate highly compelling images of specific categories, such as faces, album covers, and room interiors.



interiors. In this work, we develop a novel deep architecture and GAN formulation to effectively bridge these advances in text and image modeling, translating visual concepts from characters to pixels. We demonstrate the capability of our model to generate plausible images of birds and flowers from detailed text descriptions.

Example

this small bird has a pink breast and crown, and black primaries and secondaries.

this magnificent fellow is almost all black with a red crest, and white cheek patch.

the flower has petals that are bright pinkish purple with white stigma

this white and yellow flower have thin white petals and a round yellow stamen

Figure 1. Examples of generated images from text descriptions. Left: captions are from zero-shot (held out) categories, unseen text. Right: captions are from the training set.

Related Work

Traditionally this type of detailed visual information about an object has been captured in attribute representations - distinguishing characteristics the object category encoded into a vector (Farhadi et al., 2009; Kumar et al., 2009; Parikh & Grauman, 2011; Lampert et al., 2014),

- in particular to enable zero-shot visual recognition (Fu et al., 2014; Akata et al., 2015), and recently,
- for conditional image generation (Yan et al., 2015).

Recently, deep convolutional and recurrent networks for text have yielded highly discriminative and generalizable (in the zero-shot learning sense) text representations learned automatically from words and characters (Reed et al., 2016).

Problem Division

Sub-problems:

- first, learn a text feature representation that captures the important visual details;
- and second, use these features to synthesize a compelling image that a human might mistake for real. (using GANs)

One more issue...

However, one difficult remaining issue not solved by deep learning alone is that the distribution of images conditioned on a text description is highly multimodal, in the sense that there are very many plausible configurations of pixels that correctly illustrate the description. The reverse direction (image to text) also suffers from this problem but learning is made practical by the fact that the word or character sequence can be decomposed sequentially according to the chain rule; i.e. one trains the model to predict the next token conditioned on the image and all previous tokens, which is a more well-defined prediction problem.

Solution? - GANs. The discriminator acts as the ‘smart’ adaptive loss function.

Background

GANs:

Generative adversarial networks (GANs) consist of a generator G and a discriminator D that compete in a two- player minimax game: The discriminator tries to distinguish real training data from synthetic images, and the generator tries to fool the discriminator. Concretely, D and G play the following game on V(D,G):

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{x\sim p_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

Deep Symmetric structured joint embedding:

To obtain **a visually-discriminative vector representation of text descriptions**, we follow the approach of Reed et al. (2016) by using deep convolutional and recurrent text encoders that **learn a correspondence function with images**. The text classifier induced by the learned correspondence function ft is trained by optimizing the following structured loss:

$$\frac{1}{N}\sum_{n=1}^{N}\Delta(y_n, f_v(v_n)) + \Delta(y_n, f_t(t_n)) \tag{2}$$

where $\{(v_n, t_n, y_n) : n = 1, ..., N\}$ is the training data set, Δ is the 0-1 loss, v_n are the images, t_n are the corresponding text descriptions, and y_n are the class labels. Classifiers f_v and f_t are parametrized as follows:

$$f_v(v) = \arg\max_{y\in\mathcal{Y}} \mathbb{E}_{t\sim\mathcal{T}(y)}[\phi(v)^T\varphi(t))] \tag{3}$$

$$f_t(t) = \arg\max_{y\in\mathcal{Y}} \mathbb{E}_{v\sim\mathcal{V}(y)}[\phi(v)^T\varphi(t))] \tag{4}$$

where ϕ is the image encoder (e.g. a deep convolutional neural network), φ is the text encoder (e.g. a character-level CNN or LSTM), $\mathcal{T}(y)$ is the set of text descriptions of class y and likewise $\mathcal{V}(y)$ for images. The intuition here is that a text encoding should have a higher compatibility score with images of the correspondong class compared to any other class and vice-versa.

Matching-Aware Discriminator (GAN-CLS)

Algorithm 1 GAN-CLS training algorithm with step size α, using minibatch SGD for simplicity.

Input: minibatch images x, matching text t, mismatching $\hat{t}$, number of training batch steps S
for $n = 1$ **to** S **do**
  $h \leftarrow \varphi(t)$ {Encode matching text description}
  $\hat{h} \leftarrow \varphi(\hat{t})$ {Encode mis-matching text description}
  $z \sim \mathcal{N}(0, 1)^Z$ {Draw sample of random noise}
  $\hat{x} \leftarrow G(z, h)$ {Forward through generator}
  $s_r \leftarrow D(x, h)$ {real image, right text}
  $s_w \leftarrow D(x, \hat{h})$ {real image, wrong text}
  $s_f \leftarrow D(\hat{x}, h)$ {fake image, right text}
  $\mathcal{L}_D \leftarrow \log(s_r) + (\log(1 - s_w) + \log(1 - s_f))/2$
  $D \leftarrow D - \alpha \partial \mathcal{L}_D / \partial D$ {Update discriminator}
  $\mathcal{L}_G \leftarrow \log(s_f)$
  $G \leftarrow G - \alpha \partial \mathcal{L}_G / \partial G$ {Update generator}
end for

Model Architecture

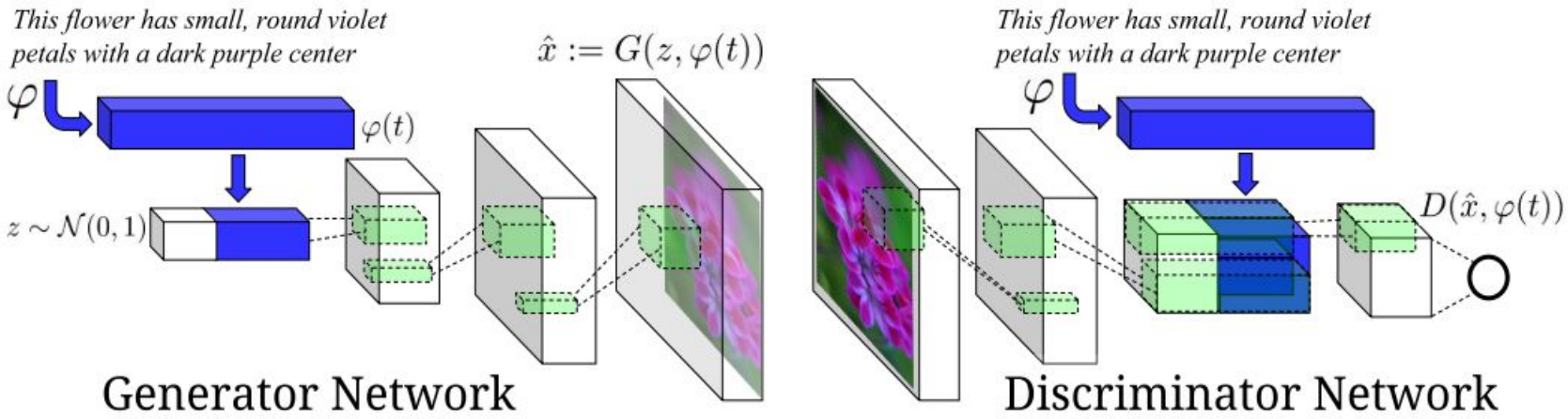


Figure 2. Our text-conditional convolutional GAN architecture. Text encoding $\varphi(t)$ is used by both generator and discriminator. It is projected to a lower-dimensions and depth concatenated with image feature maps for further stages of convolutional processing.

Learning with Manifold Interpolation (GAN-INT)

- Deep networks have been shown to learn representations in which interpolations between embedding pairs tend to be near the data manifold (Bengio et al., 2013; Reed et al., 2014).

- Motivated by this property, we can generate a large amount of additional text embeddings by simply interpolating between embeddings of training set captions.

- Critically, these interpolated text embeddings need not correspond to any actual human-written text, so there is no additional labeling cost.

Results: Birds Dataset

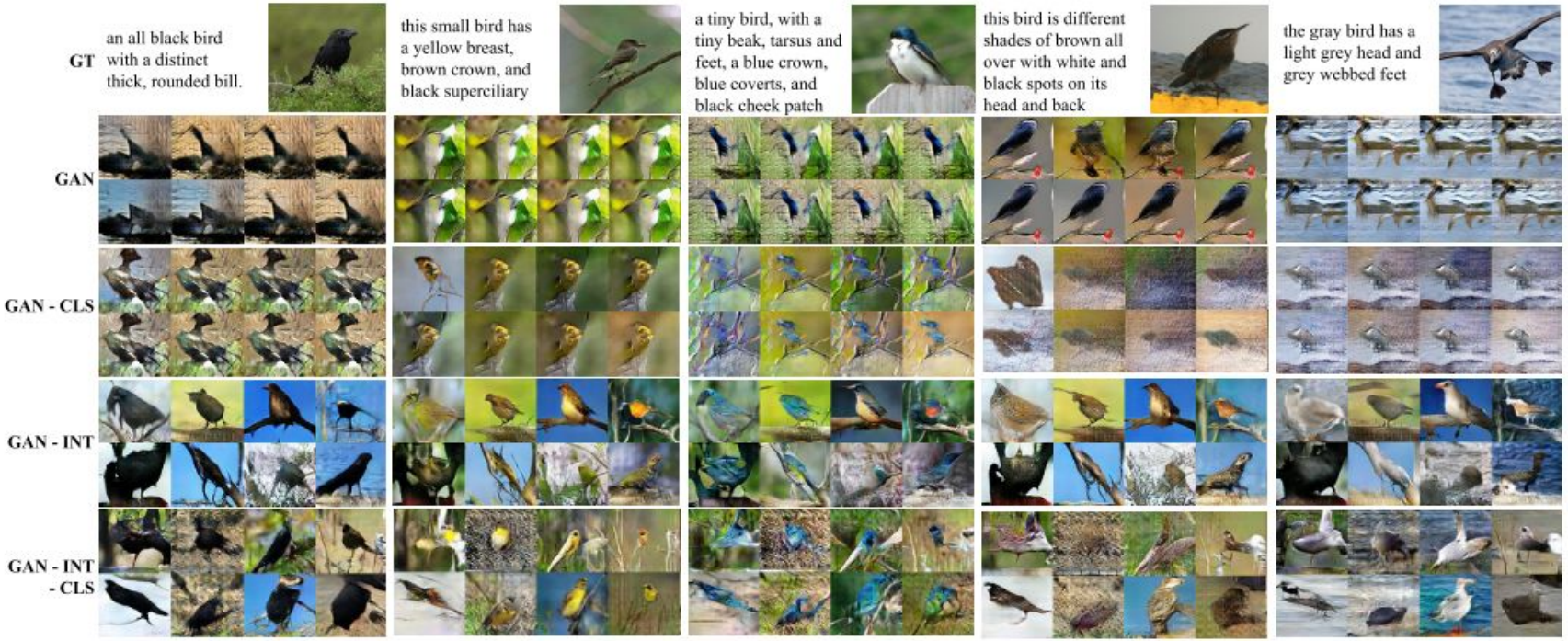


Figure 3. Zero-shot (i.e. conditioned on text from unseen test set categories) generated bird images using GAN, GAN-CLS, GAN-INT and GAN-INT-CLS. We found that interpolation regularizer was needed to reliably achieve visually-plausible results.

Results: Flower Dataset

Figure 4. Zero-shot generated flower images using GAN, GAN-CLS, GAN-INT and GAN-INT-CLS. All variants generated plausible images. Although some shapes of test categories were not seen during training (e.g. columns 3 and 4), the color information is preserved.

Thank you