Generating Images from Captions with Attention

Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba & Ruslan Salakhutdinov
University of Toronto

Abstract

Motivated by the recent progress in generative models, we introduce a model that generates images from natural language descriptions. The proposed model iteratively draws patches on a canvas, while attending to the relevant words in the description. After training on MS COCO, we compare our models with several baseline generative models on image generation and retrieval tasks. We demonstrate our model produces higher quality samples than other approaches and generates images with novel scene compositions corresponding to previously unseen captions in the dataset. For more details, visit `http://arxiv.org/abs/1511.02793`.

1 Introduction

Statistical natural image modelling remains a fundamental problem in computer vision and image understanding. Previously studied generative models of images often defined distributions that were restricted to being either unconditioned or conditioned on classification labels. In real world applications, however, images rarely appear in isolation as they are often accompanied by unstructured textual descriptions, such as on web pages and in books. The additional information from these descriptions could be used to simplify the image modelling task. Moreover, learning generative models conditioned on text also allows a better understanding of the generalization performance of the model, as we can create textual descriptions of completely new scenes not seen at training time.

In this paper, we address the problem of image generation from unstructured natural language captions. By extending the Deep Recurrent Attention Writer (DRAW) [1], our model iteratively draws patches on a canvas, while attending to the relevant words in the description. Overall, the main contributions of this work are the following: we introduce a conditional alignDRAW model, a generative model of images from captions using a soft attention mechanism. The images generated by our alignDRAW model are refined in a post-processing step by a deterministic Laplacian pyramid adversarial network [2]. We then illustrate how our method, learnt on Microsoft COCO, generalizes to captions describing novel scenarios that are not seen in the dataset.

2 Model

Our proposed model defines a generative process of images conditioned on the caption. In particular, captions are represented as a sequence of consecutive words and images are represented as a sequence of patches drawn on canvas c_t over time $t = 1, ..., T$. Our model can be viewed as utilizing the sequence-to-sequence framework [3].

2.1 Language Representation: the Bidirectional Attention RNN

Let $\mathbf{y}$ be the input caption, consisting of N words $y_1, y_2, ..., y_N$, and $\mathbf{x}$ be the output image. We obtain the caption sentence representation by first transforming each word $y_1, ..., y_N$ to a vector representation using the Bidirectional RNN. In a Bidirectional RNN, the two LSTMs process the input sequence from both forward and backward directions. They produce the hidden states sequences $[\overrightarrow{h}_1^{lang}, \overrightarrow{h}_2^{lang}, ..., \overrightarrow{h}_N^{lang}]$ and $[\overleftarrow{h}_1^{lang}, \overleftarrow{h}_2^{lang}, ..., \overleftarrow{h}_N^{lang}]$ respectively. These hidden states are then concatenated together into the final sentence representation $\boldsymbol{h}^{\boldsymbol{lang}} = [h_1^{lang}, h_2^{lang}, ..., h_N^{lang}]$, where $h_n^{lang} = [\overrightarrow{h}_n^{lang}, \overleftarrow{h}_n^{lang}], 1 \leq n \leq N$.

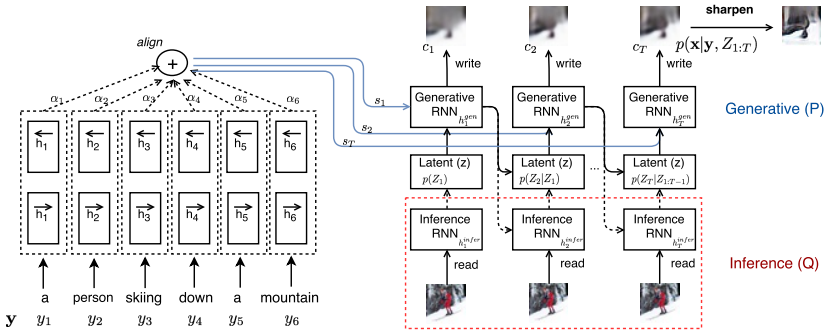


Figure 1: alignDRAW: Generative model of images conditioned on captions.

2.2 Image Modelling: the Conditional alignDRAW Network

To generate images conditioned on the caption information, we extended the DRAW network [1] to include caption representation $\boldsymbol{h^{lang}}$ at each step, shown in Figure 1. The conditional DRAW network is a stochastic recurrent neural network that consists of a set of latent variables Z_t at each time step. Unlike the original DRAW network where latent variables are independent unit Gaussians $\mathcal{N}(0, I)$, the latent variables in the proposed alignDRAW model have their mean and variance depend on the previous recurrent hidden states h^{dec}_{t-1} as in [4]. Formally, the image is generated by iteratively computing the following equations for $t = 1, ..., T$ (see Figure 1):

$$z_t \sim p(Z_t|Z_{1:t-1}) = \mathcal{N}(\mu_t(h^{dec}_{t-1}), \sigma_t(h^{dec}_{t-1})), \tag{1}$$

$$h^{dec}_t = LSTM^{dec}(h^{dec}_{t-1}, z_t, s_{t-1}), \tag{2}$$

$$s_t = align(h^{dec}_{t-1}, \boldsymbol{h^{lang}}); \qquad c_t = c_{t-1} + write(h^{dec}_t), \tag{3}$$

where $write$ and $read$ are the same attention operators as in [1]. The $align$ function is used to compute the alignment between the input caption and intermediate image generative steps [5]. Given the caption representation from the language model, $\boldsymbol{h^{lang}} = [h^{lang}_1, h^{lang}_2, ..., h^{lang}_N]$, the $align$ operator outputs a dynamic sentence representation s_t at each step through a weighted sum using alignment probabilities $\alpha_{1...N}$: $s_t = align(h^{dec}_{t-1}, \boldsymbol{h^{lang}}) = \alpha_1 h^{lang}_1 + \alpha_2 h^{lang}_2 + ... + \alpha_N h^{lang}_N$. The corresponding alignment probabilities $\alpha_{1...N}$ at each step are obtained using the caption representation $\boldsymbol{h^{lang}}$ and the hidden state of the generative model h^{dec}_t, as in [5].

2.3 Learning and Generation

Our conditional alignDRAW model is trained to maximize the variational lower bound on the log-likelihood, $\log \sum_{Z_{1:T}} p(Z_{1:T}) p(\mathbf{x} \,|\, \mathbf{y}, Z_{1:T})$. The posterior inference is approximated by an inference RNN $q(Z_{1:T} \,|\, \mathbf{y}, \mathbf{x})$ shown in the red dashed box in Figure 1. Overall, the variational objective $\mathcal{L}$ is defined with the model parameters vector θ as follows:

$$\begin{aligned}\mathcal{L}_\theta =& \mathbb{E}_{q(Z_{1:t} \,|\, \mathbf{y}, \mathbf{x})} \left[-\log p(\mathbf{x} \,|\, \mathbf{y}, Z_{1:T}) + \sum_{t=2}^{T} \mathrm{D_{KL}} \left(q(Z_t \,|\, Z_{1:t-1}, \mathbf{y}, \mathbf{x}) \,\|\, p(Z_t \,|\, Z_{1:t-1}, \mathbf{y}) \right) \right] \\ &+ \mathrm{D_{KL}} \left(q(Z_1 \,|\, \mathbf{x}) \,\|\, p(Z_1 \,|\, \mathbf{y}) \right).\end{aligned} \tag{4}$$

At the test time, images are generated by ancestral sampling the latent variables from the prior $p(Z_{1:T})$. The generator of an adversarial network, trained independently as in [2] on the residuals of a Laplacian pyramid, is used to sharpen the generated images from alignDRAW, which are often blurry. Instead of sampling from its prior, we fix the input to the adversarial generator to be the mean of the original uniform distribution. This post-processing step is a deterministic mapping which enables us to calculate the lower bound on the new log-likelihood defined at the output of the adversarial generator. Interestingly, we found that the deterministic process generates samples with much less noise than if we had sampled from the uniform distribution.

3 Experiments on MS COCO dataset

In the following subsections, we analyze both the qualitative and quantitative aspects of our model as well as compare its performance with that of other, related generative models[1]. First, we wanted to see whether the model understood one of the most basic properties of any object, the color. In

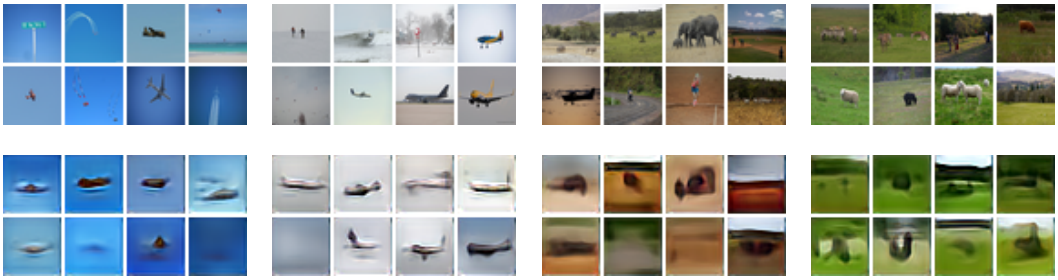
[1]To see more generated images, go to `http://www.cs.toronto.edu/~emansim/cap2im.html`

Figure 2, we generated images of school buses with four different colors: yellow, red, green and blue. Although, there are images of buses with different colors in the training set, all mentioned school buses are specifically colored yellow. Despite that, the model managed to generate images of an object that is visually reminiscent of a school bus that is painted with the specified color.

A yellow school bus parked in a parking lot. | A red school bus parked in a parking lot. | A green school bus parked in a parking lot. | A blue school bus parked in a parking lot.

Figure 2: Examples of changing the color while keeping the caption fixed. Best viewed in colour.

Apart from changing the colors of objects, we experimented with changing the background of the scene described in a caption to see whether this would result in the appropriate changes in the generated samples. The task of changing the background of an image is somewhat harder than just changing the color of an object because the model will have to make alterations over a wider visual area. Nevertheless, as shown in Figure 3, changing the skies from blue to rainy in a caption as well as changing the grass type from dry to green in another caption resulted in the appropriate changes in the generated image. The nearest images from the training set also indicate that the model was not simply copying the patterns it observed during the learning phase.



A very large commercial plane flying in blue skies. | A very large commercial plane flying in rainy skies. | A herd of elephants walking across a dry grass field. | A herd of elephants walking across a green grass field.

Figure 3: **Bottom**: Examples of changing the background while keeping the caption fixed. **Top**: The respective nearest training images based on pixel-wise L2 distance. Best viewed in colour.

Despite some success with changing colors and backgrounds in descriptions, the model struggled when the visual difference between objects was very small, such as when the objects have the same general shape and color. In Figure 4, we demonstrate that when we swap two objects that are both visually similar, for example cats and dogs, it is difficult to discriminate solely from the generated samples whether it is an image of a cat or dog, even though we might notice an animal-like shape.

The decadent chocolate desert is on the table. | A bowl of bananas is on the table. | A vintage photo of a cat. | A vintage photo of a dog.

Figure 4: Examples of changing the object while keeping the caption fixed.

During the generation step, the model mostly focused on the specific words that carried the main semantic meaning expressed in the sentences. The attention values in sentences helped us interpret the reasons why the model made the changes it did when we flipped certain words. In Figure 5 we can see that when we flipped the word "desert" to "forest", the attention over words in the sentence did not change drastically. Effectively, the model looked at "desert" and "forest" with relatively equal probability, and thus made the correct changes. In contrast, when we swap words "beach" and "sun", the model completely ignores the word "sun" in the second sentence, which gives us a more thorough understanding of why we see no visual differences between the images generated by each

caption. We also tried to analyze the way the model generated images. Unfortunately, we found that there was no connection between the patches drawn on the canvas and the words with highest attention at particular timesteps.

A rider on a blue motorcycle in the desert. | A rider on a blue motorcycle in the forest. | A surfer, a woman, and a child walk on the beach. | A surfer, a woman, and a child walk on the sun.

Figure 5: Examples of most attended words while changing the background in the caption.

3.1 Comparison With Other Models

To compare performances of different generative models, we report results on two different metrics as well as a qualitative comparison of different generative models. In Figure 6, we generated several samples from the prior of each of the current state-of-the-art generative models, conditioned on the caption "A group of people walk on a beach with surf boards".

alignDRAW | LAPGAN | Conv. VAE | Fully-conn. VAE

Figure 6: alignDRAW and three baseline models conditioned on skip-thoughts [6] displaying results from sampling caption "A group of people walk on a beach with surf boards". LAPGAN samples look more noisy and it is harder to make out definite objects, whereas the images generated by variational models trained with L2 cost function have a watercolor effect.

To compare performances of variational models, we rank the images conditioned on the captions in the test set based on the lower bound of the log-probability and then report the Precision-Recall metric (see Table 1). To avoid looping through each test image, we create a shortlist of 100 images including the correct one, based on the closest distance in the convolutional feature-space of a VGG-like model trained on the CIFAR dataset. To deal with "easy" images, we took a ratio of image likelihood conditioned on the caption to image likelihood conditioned on the mean training caption representation [7]. We found that the lower bound of the test log-probability decreased for sharpened images, which considerably hurt the retrieval.

In addition we calculate Structural Similarity Index (SSI), which incorporates luminance and contrast masking into the error calculation. The metric is also calculated on small windows of the image. We sampled 50 images from the prior of each generative model for every test caption and calculated SSI, which is reported in Table 1.

	Image Search					Image Similarity
Model	**R@1**	**R@5**	**R@10**	**R@50**	**Med r**	**SSI**
LAPGAN	-	-	-	-	-	0.08
Fully-conn. VAE (L2 cost)	1.0	6.6	12.0	53.4	47	0.156
Conv. VAE (L2 cost)	1.0	6.5	12.0	52.9	48	0.164
skipthoughtDRAW	2.0	11.2	18.9	63.3	36	0.157
noalignDRAW	2.8	14.1	23.1	68.0	31	0.155
alignDRAW	3.0	14.0	22.9	68.5	31	0.156

Table 1: Results of different models on COCO dataset (before sharpening).

References

[1] Karol Gregor et. al. DRAW: A recurrent neural network for image generation. *ICML 2015*.

[2] Emily Denton et al. Deep generative image models using a laplacian pyramid of adv. nets. *NIPS 2015*.

[3] Ilya Sutskever et al. Sequence to sequence learning with neural networks. *NIPS 2014*.

[4] Philip Bachman et al. Data generation as sequential decision making. *NIPS 2015*.

[5] D. Bahdanau, et al. Neural machine translation by jointly learning to align and translate. *ICLR*, 2015.

[6] R. Kiros et al. Skip-thought vectors. *NIPS 2015*.

[7] R. Kiros, et al. Unifying visual-semantic embeddings with neural language models. *TACL*, 2014.