Variational Encoder Decoder for Image Generation Conditioned on Captions

Nicolas Angelard-Gontier NICOLAS.ANGELARD-GONTIER@MAIL.MCGILL.CA
McGill University

Joshua Romoff JOSHUA.ROMOFF@MAIL.MCGILL.CA
McGill University

Prasanna Parthasarathi PRASANNA.PARTHASARATHI@MAIL.MCGILL.CA
McGill University

Abstract

In this project we used a Variational Encoder Decoder (VED) to generate images from text. A VED works identically to the traditional Variational Auto-Encoder (VAE); it encodes the input to a latent space, samples Gaussian variables from the latent representation, and decodes the latent variables into the output. The fundamental difference between the two models is the fact that the VED's output is not necessarily from the same representation as its input. We used the MNIST dataset to automatically create different sets of captions, which include: image labels, noisy image labels, logical operations, and basic additions. Our results show that the model is able to generate correct images on never before seen, more complex combinations of captions.

1. Literature Review

Deep neural networks have achieved significant successes in many different tasks such as speech recognition (Hinton et al., 2012), image classification (Krizhevsky et al., 2012), image captioning (Karpathy & Fei-Fei, 2017), and machine translation (Bahdanau et al., 2014). However, most of these past successes have come from discriminative models, whereas generative models have only recently risen to the forefront of the Deep Learning community.

There has been a vast amount of Deep Learning research directed towards the understanding of text and images. The first of which (the most simple) is the classification setting; where the model must identify the correct label for each image. Due to the success of deep discriminative models (Krizhevsky et al., 2012), image classification is largely considered solved. Caption generation is a popular task that has also achieved great successes (Karpathy & Fei-Fei, 2017). One of the reasons for its recent popularity is due to the vast amount of available labeled data (e.g the MSCOCO dataset (Lin et al., 2014)). The approaches to model a joint distribution over text and images motivated further to work in the reverse direction: generating an image given its textual description. This challenging task involves language modeling and conditional image synthesis.

Image generation from text is a task that conditions a model to learn a joint distribution between pixel values and a text description of visual features. The base components that are present in many modern approaches are the following; text embeddings that map from text to some latent vector representation, a Recurrent Neural Network (RNN) to encode the embeddings by capturing time dependencies, and a decoder that maps from latent variables to images. In general, the word embeddings can be either learned or pretrained on some large corpus (e.g word2vec (Mikolov et al., 2013)), the (RNN) can be either unidirectional or bidirectional, and the decoder can either be a simple feed forward neural network or a deconvolutional neural network.

One very popular method for image generation is to use a Variational Auto-Encoder (Kingma & Welling, 2013). VAEs learn to encode and decode their inputs, while also approximating the posterior distribution as a mapping from a standard normal distribution. Thus, the encoders job is to both help the decoder reconstruct the input distribution while also minimizing the distance between its encoding and the normal distribution.

There are several recent works that use VAEs to generate images from captions. For example, (Pu et al., 2016) used a Deconvolutional Neural Network to form the mapping from latent variables to the output image. Another example is the work of (Mansimov et al., 2015), where they it-

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48.

eratively generate fragments of the image by using a VAE combined with a bidirectional RNN as an attention mechanism over the captions.

It is worth mentioning that there exists other popular approaches in the Deep learning literature to handle these generative tasks. One particular method is to formulate the image generation as a minimax problem using Generative Adversarial Networks (GANs) (Goodfellow et al., 2014). The main idea of GANs is simple; train a model to generate data by fooling a second model (the discriminator) that attempts to discriminate between true samples and the generated (fake) samples. In the case of image generation from captions, the generator generates images from both a gaussian noise vector and a caption. Using GANs in this context has resulted in fairly sharp generated images (Reed et al., 2016; Zhang et al., 2016).

However, GANs are notoriously difficult to train due to mode collapse and non-convergence brought on by the fact that there are two models to train simultaneously (Goodfellow, 2017) (although newer methods claim to alleviate the problem, e.g WGAN (Arjovsky et al., 2017)). For this reason, in this paper we will focus mainly on the Variational Auto-Encoder. We provide a more formal description of the VAE in the following subsections.

1.1. Variational Auto Encoder Background

In general, a VAE consists of an encoder, parameterized by θ, that takes in as input $\mathbf{x} \in \mathbb{R}^n$ and projects it to a lower dimensional space $\mathbf{z} \in \mathbb{R}^d$ and a decoder, parameterized by ϕ, that does the inverse operation of the encoder. The encoder learns an effective compression through this bottleneck projection, $q_\theta(\mathbf{z} \mid \mathbf{x})$. The distribution q_θ is conditioned to be a standard Gaussian that serves as a regularizer in learning the distribution. The decoder learns $p_\phi(\mathbf{x} \mid \mathbf{z})$ to decode $\mathbf{x}$ from a sample from q_θ. By learning to map $\mathbf{z}$ to a standard Gaussian during training, a sample from a standard Gaussian can be used as $\mathbf{z}$, input to the decoder, to decode $\mathbf{x}$, thus learning the distribution over $\mathbf{x}$ effectively in a lower dimensional latent space $\mathbf{z}$.

VAEs are trained by minimizing the negative log-likelihood regularized with a KL-divergence factor as shown in the equation below:

$$L = \mathbb{E}_{z\sim q_\theta}\left[\log p_\phi(\mathbf{x} \mid \mathbf{z})\right] + KL\left[q_\theta(\mathbf{z} \mid \mathbf{x}) \,||\, p(\mathbf{z})\right] \quad (1)$$

with $p(\mathbf{z})$ specified as a standard Gaussian. The KL loss, hence, penalizes the encoder if p_θ is different from standard Gaussian thus enabling it to be *sufficiently diverse*.

1.2. Graphical Model Perspective

A Variational Auto Encoder learns a joint distribution between the output and the latent variables that factorizes as, $p(x, z) = p(x|z)p(z)$. The decoder network decodes the latent variable, $z_i \sim p(z)$, to a vector in the image space during the generative process.

The goal of a VAE is to infer good values of z that can be decoded into the right image. By Bayes law,

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)} \quad (2)$$

Computing the posterior requires exponential time as the denominator $p(x)$ in Equation 2 needs to be computed over all possible configurations. To avoid this, the model assumes a parameterized distribution $q_\lambda(z|x)$, where λ indicates a family of distributions. We consider the Gaussian family of distributions which assumes a mean μ_{x_i} and a variance $\sigma^2_{x_i}$ for every datapoint and we try to directly minimize the KL-Divergence between q_λ and $p(z|x)$:

$$KL\left(q_\lambda(z)||p(z|x)\right) = E_q\left[\log q_\lambda(z)\right] - E_q\left[\log p(x, z)\right] + \log p(x) \quad (3)$$

Minimizing KL loss as in Equation 3 requires to compute the intractable $p(x)$. To get around this, we consider Jensen inequality on the factorization of $p(x)$,

$$\begin{aligned}
\log p(x) &= \int_z \log p(x, z) \\
&= \int_z \log p(x, z) \frac{q(z)}{q(z)} \\
&= \log E_q\left[\frac{p(x, z)}{q(z)}\right] \\
&\geq E_q\left[\log(p(x, z))\right] - E_q\left[\log q(z)\right]
\end{aligned}$$

This is the Evidence Lower Bound (ELBO). KL loss can be now re-written as,

$$KL = -ELBO + \log p(x)$$

As the $\log p(x)$ term is independent of q, maximizing ELBO directly corresponds to minimizing KL. This is the loss term in training variational models and our proposed architecture uses this to optimize the image generation task.

2. Model Description

We use an encoder-decoder architecture (see Figure 1) to model the posterior distribution $P(z|x)$. The encoder (a Recurrent Neural Network) takes in as input a sequence of word embeddings $(x_{1:t})$ and encodes it to a latent space (z). The variational objective is that the conditional distribution over z is not easy to sample from. Hence, the model uses a standard Gaussian with identity covariance matrix and zero mean to model the conditional distribution over

the latent space. This provides an amenable distribution to be sampled from during testing.

The model is trained end-to-end using Equation 1. The first term in the expression tries to minimize the error in reconstruction from the latent space, and the second term is the variational lower bound, which approximates the posterior to standard Gaussian.

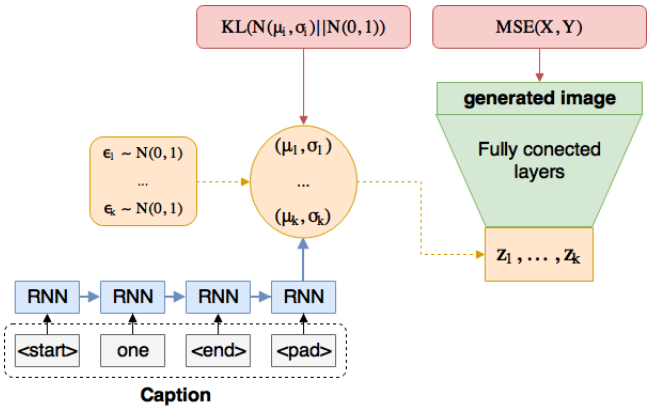

Figure 1. Variational Encoder Decoder Model

3. Methodology

In this section we present the data we used and provide hyper-parameter settings.

3.1. Data description

We used MNIST database for our experiments. MNIST is a collection of handwritten digit images. Each image is a 28×28 binary image with corresponding class label. We manually defined a variety of captions ranging from easy to more difficult ones:

1. image label as caption, ie: "<start> *five* <end>" for all images.

2. random sentences with one label in it, ie: "<start> this is a black *five* on a white background <end>". We have a total of 12 different sentences, and for each image, we sample one of them to be the caption.

3. logical captions with multiple numbers in it, ie: "<start> min nine *five* seven <end>" should generate an image of a five. We considered 'min' and 'max' operators.

4. operation captions where we write a sum of multiple numbers that represent the image label, ie: "<start> three plus two <end>" should generate an image of a five.

3.2. Implementation details

We started to experiment with simple captions to make sure our model was bug-free [1] (caption type 1), and subsequently tried more and more complex captions. The model was performing relatively well on all types of captions so we decided to focus on the operational captions (type 4). We used Pytorch (Paszke et al., 2017) as our Deep Learning library.

We divided the MNIST dataset into a training set, validation set, and test set. We then hand defined 13 different training and validation captions for each digit class. In particular we considered only one operation between two digits for each number to generate (*'x plus y'*), the actual number to generate (*'x'*), and the decomposition of that number into sums of 'one's (*'one plus one ... plus one'*). At test time the captions were much more complex and longer. For the model to perform well in this more complex setting, it needed to learn the "compositionallity" of simple addition or subtraction.

In order to better evaluate our generative model we also trained a classic Convolutional Neural Network to predict the digit label given an MNIST image. We trained on original MNIST images up to an accuracy of 99%. We report, in the section below, the accuracy of this classifier (keeping it fixed) on the task of labeling generated images. We expect to see an increase in its accuracy as the VED model learns to generated better and better images.

The final model that we used for our experiments was quite small: fixed word embeddings of dimension 5 were chosen, a simple uni-directional Gated Recurrent Network (Chung et al., 2014) as the encoder and 10 gaussian distributions were sampled before getting decoded by a one-layer feed-forward neural network. We trained the model up until there was no improvement in the reconstruction loss on the validation set.

4. Results

We present several preliminary results in this section. First if we consider Figure 2, what we see is that both the training reconstruction error and the validation reconstruction error decreases monotonically over time. This can be expected since both the training and validation set use the same distribution over captions. The training time for this experiment was around 10 minutes on a system equipped with a Titan X.

The result of feeding our model's generated images into a pre-trained discriminator (described in the previous section) is provided in Figure 3. Here we see both a positive

[1] The code is available at `https://github.com/ppartha03/PGM-Project`

and a negative result. On the positive side, it is clear that the discriminator improves over the course of training (definitely better than random (10%)). On the negative side, however, the final accuracy is far from optimal. The negative result can be explained by the fact that we pre-trained the classifier on clear and crisp images from the original data-set, whereas our model generates admittedly noisy images.

Finally, we present the result of our model's capacity to generalize to unseen captions in Figure 4. Each row represents one caption from the test set. The left most column is the caption and the subsequent columns are different samples from our model. We recall that our model only ever trained on captions with a single arithmetic operation. While not perfect, the samples are mostly discriminable for humans, although we can see that are model has some trouble generating 7s that look like 1s and 5s that look like 3s.

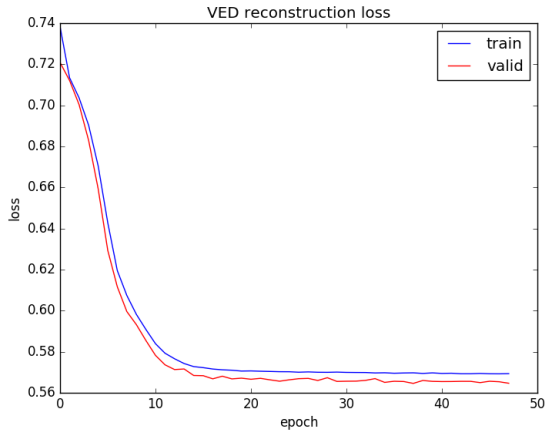

Figure 2. Variational Encoder Decoder reconstruction loss

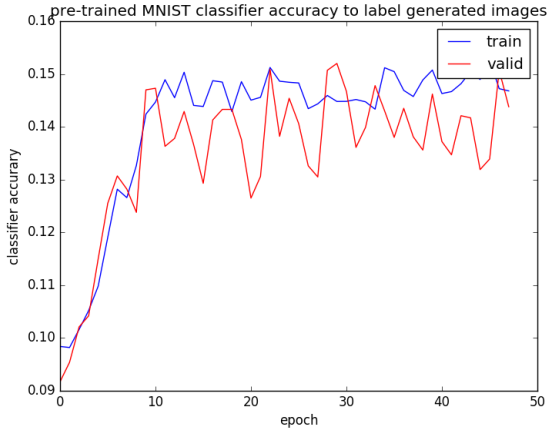

Figure 3. Pre-trained MNIST classifier accuracy on generated images

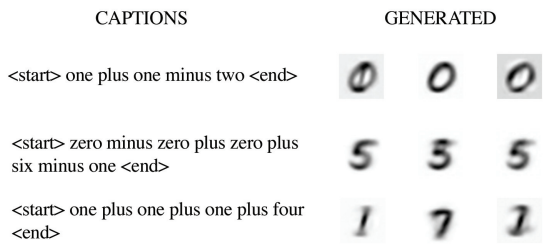

Figure 4. Generated images from test set captions

5. Discussion

In this paper we explored using a form of VAE for caption to image generation. In order to highlight its generative power, we created our very own variant of MNIST that involved generating images from captions that were based on simple arithmetic operations. We were able to show that using a very simple architecture for our VED (an RNN as the encoder and a fully connected network as the decoder), we can successfully generalize to unseen captions.

After conducting our experiment we found some similar tasks to our arithmetic image generation in the literature. (Hoshen & Peleg, 2015) defined a task where the goal was to do some arithmetic operation on two images that each contained a sequence of several hand-written MNIST digits. As well, (Jaderberg et al., 2015) conducted an experiment, in their appendix, where the model sees two MNIST images and must output the numeric sum (not as text but as an integer). This is fairly close to the inversion of our experiment.

As future work we would like to experiment with using deconvolution layers instead of fully connected layers, as in (Pu et al., 2016). We would also be interested in utilizing the attention mechanism over captions to iteratively build up our image, that was introduced in (Mansimov et al., 2015). It would also be interesting to try more complex arithmetic operations to see how well our model scales on that front.

Acknowledgments

The authors gratefully acknowledge Simon Lacoste-Julien for the great class and Sarath Chandar for being a great TA.

References

Arjovsky, Martin, Chintala, Soumith, and Bottou, Léon. Wasserstein generative adversarial networks. In Precup, Doina and Teh, Yee Whye (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 214–223, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

Bahdanau, Dzmitry, Cho, Kyunghyun, and Bengio, Yoshua. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. URL http://arxiv.org/abs/1409.0473.

Chung, Junyoung, Gulcehre, Caglar, Cho, KyungHyun, and Bengio, Yoshua. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014. URL https://arxiv.org/abs/1412.3555.

Goodfellow, Ian, Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, and Bengio, Yoshua. Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. Curran Associates, Inc., 2014. URL http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

Goodfellow, Ian J. NIPS 2016 tutorial: Generative adversarial networks. *CoRR*, abs/1701.00160, 2017. URL http://arxiv.org/abs/1701.00160.

Hinton, Geoffrey, Deng, Li, Yu, Dong, Dahl, George, Mohamed, Abdel-rahman, Jaitly, Navdeep, Senior, Andrew, Vanhoucke, Vincent, Nguyen, Patrick, Kingsbury, Brian, and Sainath, Tara. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 29:82–97, November 2012.

Hoshen, Yedid and Peleg, Shmuel. Visual learning of arithmetic operations. *CoRR*, abs/1506.02264, 2015. URL http://arxiv.org/abs/1506.02264.

Jaderberg, Max, Simonyan, Karen, Zisserman, Andrew, and Kavukcuoglu, Koray. Spatial transformer networks. *CoRR*, abs/1506.02025, 2015. URL http://arxiv.org/abs/1506.02025.

Karpathy, Andrej and Fei-Fei, Li. Deep visual-semantic alignments for generating image descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):664–676, April 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2598339.

Kingma, Diederik P. and Welling, Max. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013. URL http://arxiv.org/abs/1312.6114.

Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012.

Lin, Tsung-Yi, Maire, Michael, Belongie, Serge J., Bourdev, Lubomir D., Girshick, Ross B., Hays, James, Perona, Pietro, Ramanan, Deva, Dollár, Piotr, and Zitnick, C. Lawrence. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. URL http://arxiv.org/abs/1405.0312.

Mansimov, Elman, Parisotto, Emilio, Ba, Lei Jimmy, and Salakhutdinov, Ruslan. Generating images from captions with attention. *CoRR*, abs/1511.02793, 2015. URL http://arxiv.org/abs/1511.02793.

Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Greg, and Dean, Jeffrey. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013. URL http://arxiv.org/abs/1310.4546.

Paszke, Adam, Gross, Sam, Chintala, Soumith, Chanan, Gregory, Yang, Edward, DeVito, Zachary, Lin, Zeming, Desmaison, Alban, Antiga, Luca, and Lerer, Adam. Automatic differentiation in pytorch. 2017.

Pu, Yunchen, Gan, Zhe, Henao, Ricardo, Yuan, Xin, Li, Chunyuan, Stevens, Andrew, and Carin, Lawrence. Variational autoencoder for deep learning of images, labels and captions. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 2352–2360, 2016.

Reed, Scott E., Akata, Zeynep, Yan, Xinchen, Logeswaran, Lajanugen, Schiele, Bernt, and Lee, Honglak. Generative adversarial text to image synthesis. *CoRR*, abs/1605.05396, 2016. URL http://arxiv.org/abs/1605.05396.

Zhang, Han, Xu, Tao, Li, Hongsheng, Zhang, Shaoting, Huang, Xiaolei, Wang, Xiaogang, and Metaxas, Dimitris N. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. *CoRR*, abs/1612.03242, 2016. URL http://arxiv.org/abs/1612.03242.