A summary of novel advancements in generative modeling

Raja Haseeb Ur Rehman
Department of Aerospace
KAIST
South Korea
rajahaseeb147@kaist.ac.kr

Abstract

A Generative Model is a powerful way of learning any kind of data distribution using unsupervised learning and it has achieved tremendous success in just a few years. All types of generative models aim at learning the true data distribution of the training set to generate new data points with some variations. In this summary paper, some novel generative modeling concepts have been presented.

1 Introduction

In recent years, deep learning has become immensely popular and shown great success in a large number of applications including image processing, object detection [7], text mining, social media recommendations, speech recognition [4] and so forth. Furthermore, many recent works show that neural networks can also be applied in the domain of natural language processing (NLP). In the field of statistical machine learning, promising results have been shown by deep neural networks [12].

Aside from that, deep generative models have also improved a lot in recent years [2, 14, 8]. The samples generated by these models are hard to distinguish from real data. The goal of these models is to learn the true data distribution of the training set and then generate new data points with some variations. For this, we can leverage the power of neural networks to learn a function that can approximate the model distribution to the true distribution. Two of the most commonly used and efficient approaches are Variational Autoencoders (VAE) [6] and Generative Adversarial Networks (GAN) [5]. Some novel approaches in Encoder-Encoder and GAN networks are summarized in this paper.

2 Summary

There are four papers included in this survey. First one is related to the task of STM by Cho, Kyunghyun, et al. [3] . Second paper by Rifai, Salah, et al. [10] introduces some improvement on feature extraction level for the existing auto-encoder networks. Then we have two papers for generative modeling including a paper on VQ-VAE by ALi Razavi et al. [9] and the second famous paper by Martin Arjovsky et al. [1], which improves the existing GAN model.

2.1 Learning phrase representations using RNN encoder-decoder for statistical machine translation

RNN encoder-decoder model first proposed by Cho et al in 2014. It uses two recurrent neural networks. One encodes the input sequence into a fixed-length vector representation and another decodes that representations into another sequence of symbols. They are jointly trained to increase the conditional probability of the target sequence given the input sequence. In addition to standard log loss of recurrent neural network using conditional probabilities of phrase pairs computed by RNN encoder-decoder found to improve empirical performance. The paper has shown that Encoder-Decoder model learns a semantically and syntactically meaningful representation of linguistic phrases.

The paper described that it uses a novel hidden unit that is empirically evaluated on the task of translating from English to French. We train the model to learn the translation probabilities of an English phrase to a corresponding French phrase. The model is then used to part of a standard phrase-based SMT system by scoring phrase pairs in the phrase table. Encoder-Decoder model is better in capturing linguistic regularities in the phrase table, indirectly explaining the quantitative improvement in the overall translation performance. It also learns a continuous-space representation of the phrase that preserves both the semantic and syntactic structure of the phrase.

One important difference between the proposed model and approach used in citech2014autoencoder is that it can naturally distinguish between sequences that include the same words but in a different order. Also, in prior approaches, the maximum length of the input phrase was always fixed. The proposed RNN Encoder-Decoder however, is well-suited for applications where the length of input phrases increases or we apply neural networks to other variable-length sequence data.

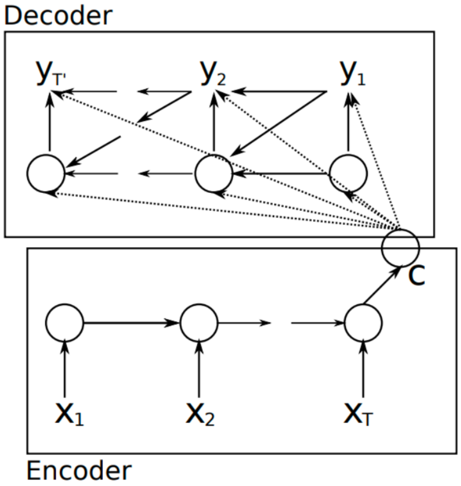


Figure 1: An illustraion of the proposed RNN Encoder-Decoder [3].

2.1.1 Results

The system achieved a BLEU score of 33.87, which is a good score compared to the baseline score developed with a statistical machine translation system of 33.30. The best performance was achieved when the phrase score from RNN Encoder-Decoder was combined with continuous space language models (CSLM) [11].

2.1.2 Remarks

RNN encoder-decoder model can learn from arbitrary length sequences and generate a target sequence of arbitrary length. The model is very successful either predicting a target sequence based on an input sequence or output a probability score given an input and output pair. When tried to score for machine translation task out of phrase mapping table model demonstrated capability to understand linguistic regularities and able to propose well-formed target phrases. The current state of the art language model and many NLP tasks are based on the premise of this encoder-decoder model including machine translation and sentiment analysis. Importantly, the Cho Model is used only to score candidate translations and is not used directly for translation like the Sutskever model [13]. Although extensions to the work to better diagnose and improve the model do use it directly and alone for translation.

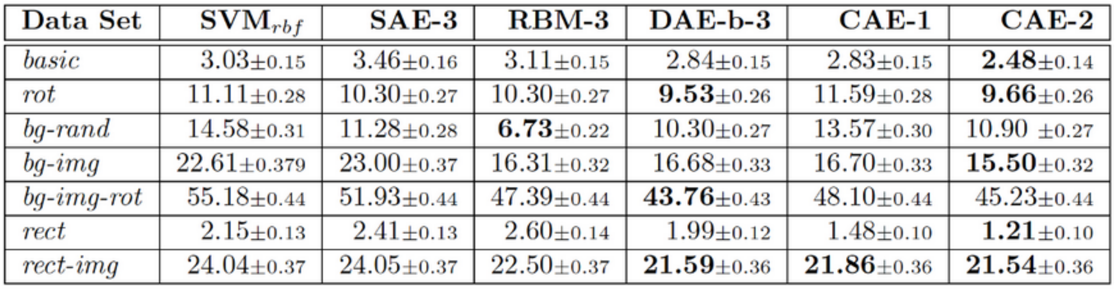
2.2 Contractive auto-encoders: Explicit invariance during feature extraction

According to this paper, carefully created penalty term can result in extracting more useful and effective features that give insight to the given data. The penalty term invented by the authors of this paper makes the auto-encoders learned features to be locally invariant without any preference for particular directions. (they obtain invariance in the directions that make sense in the context of the given training data, i.e., the variations that are present in the data should also be captured in the learned representation, but the other directions may be contracted in the learned representation). This penalty term corresponds to the Frobenius norm of the Jacobian matrix of the encoder.

Additionally, one very interesting question the authors asked is the notion, how can we extract robust features? (aka features that are robust to small changes in the given input). The way they did it by adding a penalty term that is sensitive to the given input, and as the network trains, it's objective is to make that sensitivity smaller and smaller.

2.2.1 Results

Results show that 2 layer CAE is even able to outperform other types of network that have three layers



Data Set	SVM_{rbf}	SAE-3	RBM-3	DAE-b-3	CAE-1	CAE-2
basic	3.03±0.15	3.46±0.16	3.11±0.15	2.84±0.15	2.83±0.15	**2.48**±0.14
rot	11.11±0.28	10.30±0.27	10.30±0.27	**9.53**±0.26	11.59±0.28	**9.66**±0.26
bg-rand	14.58±0.31	11.28±0.28	**6.73**±0.22	10.30±0.27	13.57±0.30	10.90 ±0.27
bg-img	22.61±0.379	23.00±0.37	16.31±0.32	16.68±0.33	16.70±0.33	**15.50**±0.32
bg-img-rot	55.18±0.44	51.93±0.44	47.39±0.44	**43.76**±0.43	48.10±0.44	45.23±0.44
rect	2.15±0.13	2.41±0.13	2.60±0.14	1.99±0.12	1.48±0.10	**1.21**±0.10
rect-img	24.04±0.37	24.05±0.37	22.50±0.37	**21.59**±0.36	**21.86**±0.36	**21.54**±0.36

Figure 2: Comparison of stacked CAE with 1 and 2 layers with other 3-layer stacked models [10]..

2.2.2 Remarks

Authors show that contractive auto-encoders (especially when they are stacked in a way similar to RBMs in a deep belief net) learn good models of high-dimensional data (such as images), and that these models can be used to obtain good representations for classification tasks. However, with further work, good quality samples can also be obtained from the model.

One thing to notice is that implementation of CAE might be little complex compared to denoising autoencoder (DAE) [15] which is very simple and require only adding one or two lines of code to normal autoencoder. Also there is no need to compute Jacobian for hidden layer in denoising autoencoder. On the other hand, CAE can use second order optimizers, and might be more sable than DAE.

2.3 Generating diverse high-fidelity images with VQ-VAE-2

VQ-VAE is a Variational AutoEncoder that uses as its information bottleneck a discrete set of codes, rather than a continuous vector. That is: the encoder creates a downsampled spatial representation of the image, wherein each grid cell of the downsampled image, the cell is represented by a vector. But, before that vector is passed to the decoder, it's discretized, by (effectively) clustering the vectors the network has historically seen, and substituting each vector with the center of the vector it's closest to. This has the effect of reducing the capacity of your information bottleneck, but without just pushing your encoded representation closer to an uninformed prior.

The part of the model that got a (small) upgrade in this paper is the prior distribution model that's learned on top of these latent representations. The goal of this prior is to be able to just sample images, unprompted, from the distribution of latent codes. Once we have a trained decoder, if we give it a grid of such codes, it can produce an image. But these codes aren't one-per-image, but rather a grid of many codes representing features in a different part of the image. To generate a set of codes corresponding to a reasonable image, we can either generate them all at once, or else (as this paper does) use an autoregressive approach, where some parts of the code grid are generated, and then subsequent ones conditioned on those. In the original version of the paper, the autoregressive model used was a PixelCNN. In this paper, the authors took inspiration from the huge rise of self-attention in recent years and swapped that operation in place of the convolutions. Self-attention has the nice benefit that you can easily have a global receptive range (each region being generated can see all other regions) which you'd otherwise need multiple layers of convolutions to accomplish.

2.3.1 Results

The fidelity of best class conditional samples by proposed method are competitive with the state of the art GANs, with broader diversity in several classes, contrasting the new method agaisnt the known limitations of GANs.

2.3.2 Remarks

The authors extend the originally proposed VQ-VAE model to learn two (top and bottom) level hierarchies of images. The only con of the model is that post-hoc PixelCNN (or PixelSnail in this paper) needs to be used to learn the prior over discrete codes to sample images at generation time. Although the authors claim that the model generates diverse and high quality looking it would be great to put some quantitative numbers on it. Doing with side-by-side samples from BigGAN and Hierarchical VQ-VAE and asking people to rate which models generated samples they prefer. As well as it would be great to see the nearest neighboring training images from the dataset according to the closest distance in the embedding space.

2.4 Wasserstein GAN

The Wasserstein GAN is an extension to the generative adversarial network that both improves the stability when training the model and provides a loss function that correlates with the quality of generated images. It is an important extension to the GAN model and requires a conceptual shift away from a discriminator that predicts the probability of a generated image being "real" and toward the idea of a critic model that scores the "realness" of a given image. This conceptual shift is motivated mathematically using the earth mover distance, or Wasserstein distance, to train the GAN that measures the distance between the data distribution observed in the training dataset and the distribution observed in the generated examples.

Instead of using a discriminator to classify or predict the probability of generated images as being real or fake, the WGAN changes or replaces the discriminator model with a critic that scores the realness or fakeness of a given image. This change is motivated by a mathematical argument that training the generator should seek a minimization of the distance between the distribution of the data observed in the training dataset and the distribution observed in generated examples. The argument contrasts different distribution distance measures, such as Kullback-Leibler (KL) divergence, Jensen-Shannon (JS) divergence, and the Earth-Mover (EM) distance, referred to as Wasserstein distance.

2.4.1 Remarks

The benefit of the WGAN is that the training process is more stable and less sensitive to model architecture and choice of hyperparameter configurations. Importantly, the Wasserstein distance has the properties that it is continuous and differentiable and continues to provide a linear gradient, even after the critic is well trained. Perhaps most importantly, the loss of the discriminator appears to relate to the quality of images created by the generator.

Specifically, the lower the loss of the critic when evaluating generated images, the higher the expected quality of the generated images. This is important as unlike other GANs that seek stability in terms of finding an equilibrium between two models, the WGAN seeks convergence, lowering generator loss.

The difficulty in WGAN is to enforce the Lipschitz constraint. Clipping is simple but it introduces some problems. The model may still produce poor quality images and does not converge, in particular when the hyperparameter c is not tuned correctly. The model performance is very sensitive to this hyperparameter.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis, 2018.

[3] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014.

[4] G. E. Dahl, D. Yu, L. Deng, and A. Acero. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1):30–42, 2012.

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[6] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[8] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.

[9] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2, 2019.

[10] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. 2011.

[11] Holger Schwenk. Continuous space language models. *Computer Speech & Language*, 21(3):492–518, 2007.

[12] Holger Schwenk. Continuous space translation models for phrase-based statistical machine translation. In *Proceedings of COLING 2012: Posters*, pages 1071–1080, Mumbai, India, December 2012. The COLING 2012 Organizing Committee.

[13] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks, 2014.

[14] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio, 2016.

[15] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(Dec):3371–3408, 2010.