Brief summary and detailed critique on the paper: Pixel Recurrent Neural Networks

Abdullah Faiz Ur Rahman Khilji
Department of Computer Science and Engineering
National Institute of Technology Silchar
abdullahkhilji.nits@gmail.com

Abstract

The paper Pixel Recurrent Neural Network by Google DeepMind is thoroughly analyzed. A detailed critique is given of the aforementioned paper which presents a generative model for natural images that sequentially predicts pixels in images along the two spatial dimensions. The method models a discrete probability distribution and proposes a scalable solution for the unsupervised learning problem, curating results consistent with the proposed theory.

1 Introduction

Generative image modeling is an unsupervised learning problem wherein a probability density approach is utilized. The given work employs advanced two-dimensional RNNs, one being the Row LSTM layer wherein the convolution is applied along each row and the other being the Diagonal BiLSTM layer in which the convolution is applied along the diagonals.

A second simplified architecture of PixelCNN is also proposed with a fixed dependency range, using Masked convolutions. The PixelCNN preserves the spatial resolution of the input throughout the layers and outputs a conditional distribution at each location.

Modeling the pixels as discrete values using a multinomial distribution implemented using softmax provides with both representational and training advantages for the model.

2 Overview

The aim of the paper Pixel RNN is to estimate a distribution over natural images which can easily be used to calculate the feasibility of images along manageable lines along with generating newer ones. The proposed network predicts the conditional distribution by scanning each pixel at a time, garnering information from it.

For the pixel generation process, two-dimensional LSTM network was proposed that takes all the pixel inputs in a serialized fashion.

2.1 Generating Image Pixel by Pixel

The probability function $p(x)$ is defined as:

$$p(\mathbf{x}) = \prod_{i=1}^{n^2} p\left(x_i | x_1, \ldots, x_{i-1}\right)$$

Where $x_1, \ldots, x_{n^2}$ are the image pixels and the probability of the *i-th* pixel x_i given all the

previous pixels is depicted by $p\left(x_i|x_1,\ldots,x_{i-1}\right)$

Each pixel x_i is in turn determined by the three RGB color channels, and thus the distribution $p\left(x_i|\mathbf{x}_{<i}\right)$ is rewritten as follows:

$p\left(x_{i,R}|\mathbf{x}_{<i}\right)p\left(x_{i,G}|\mathbf{x}_{<i},x_{i,R}\right)p\left(x_{i,B}|\mathbf{x}_{<i},x_{i,R},x_{i,G}\right)$

The training on the pixel values is computed parallelly while the image pixel generation is sequential.

2.2 Pixel Recurrent Neural Networks

2.2.1 Row LSTM

Here the convolution is performed using a one-dimensional convolution, computing features for the whole row at once. The layer captures roughly a triangular context above the pixel under consideration.

Even though row LSTM is computationally less intense it does not takes the complete context into consideration while training.

2.2.2 Diagonal BiLSTM

It is designed to undertake computation diagonally thus incorporating the whole context along with parallelizing the process.

2.2.3 Residual Connections

The work involves training PixelRNNs of up to twelve layers of depth. Thus to involve ease of training and optimization, residual connections are used.

2.2.4 PixelCNN

PixelCNN uses multiple convolution layers to preserve the spatial resolution, while masks are used for proper conditioning and to avoid usage of future contexts.

2.2.5 Multi-Scale PixelRNN

It is composed of a singular unconditional PixelRNN and several conditional ones. The unconditional network generates a smaller $s \times s$ image which is then fed into its conditional counterpart which generates a larger $n \times n$ image.

The conditional network is similar to the standard PixelRNN, but each of its layers is biased with an upsampled version of the small $s \times s$ image.

Upsampling Process: An enlarged feature map of size $c \times n \times n$ is constructed, where c is the features in the output map.

Biasing Process: For each layer in the conditional PixelRNN a simple mapping using 1×1 unmasked convolution of $c \times n \times n$ to $4h \times n \times n$ is added to the input-to-state map of the corresponding layer.

2.2.6 Masked Convolutions

Masking is a method by which we can prevent the information flow from the future pixels into those which we would be predicting. To implement this we just need to zero out those weights and thus, that information would never be taken into consideration.

3 Experiments

3.1 Evaluation

All the models are evaluated on the log-likelihood loss function from a discrete distribution. For MNIST the negative log-likelihood in *nats* was reported. For CIFAR-10 and ImageNet, the same was reported in *bits* per dimension. The discrete normalization was done by the dimensionality of the images.

3.2 Training

The models were trained using the torch toolbox on GPUs. RMSprop proved to be the best optimizer. For smaller datasets of MNIST and CIFAR-10, the batch size of 16 images proved to be optimal, whereas for ImageNet data batch size of 64 (on 32×32 and 32 for 64×64) was chosen.

3.3 Residual Connections

Use of Residual connections is as effective as using skip connections and using both reinforces the advantage. The performance of Row LSTM increases on increasing network depth.

3.4 Results

- On the CIFAR-10 dataset without data-augmentation, the Diagonal BiLSTM gave the best performance followed by Row LSTM and PixelCNN.
- This can be deduced from the fact that Diagonal LSTM has the highest context under consideration followed by Row LSTM and PixelCNN. The Row LSTM has a partially occluded view whereas PixelCNN has the fewest pixels under consideration.
- This also concludes that having a greater context window is important. Thus, we can say that the results are consistent with the proposed idea.
- This work is the only one on the ImageNet dataset, hence providing new benchmarks.

4 Merits

- This work improves upon deep RNNs as generative models for natural images.
- Two novel two-dimensional layers, the Row LSTM, and the diagonal BiLSTM are proposed.
- Previous approaches used a continuous distribution for the pixel values, whereas this work provides a discrete distribution $p(x)$ for each conditional distribution. The distribution is then modeled with a softmax layer providing the advantage of being multimodal. Experimentally, this distribution is found easy to be learned and produces better performance compared to the continuous distribution.
- Masked convolutions allow for full dependencies between color channels.
- Residual Connections is advantageous compared to previous approaches that involve the use of gating along the depth of RNN as it does not require additional gates.
- Given the scalable nature of the model, larger data will significantly improve results.
- The PixelCNN proved to be the fastest architecture.