Representation Learning with Contrastive Predictive Coding

Md Mahfuzur Rahman
TReNDS, GSU, GATech, Emory University

Paper Details

Paper: Representation Learning with Contrastive Predictive Coding

Aaron van den Oord , Yazhe Li, Oriol Vinyals
DeepMind, Google.

The paper can be found here.

- Learning representation of by predicting future in latent space
- c_t and x_{t+k} are conditionally dependent on shared high-level latent info.

General Intuition

Basic Intuition

Speech representation

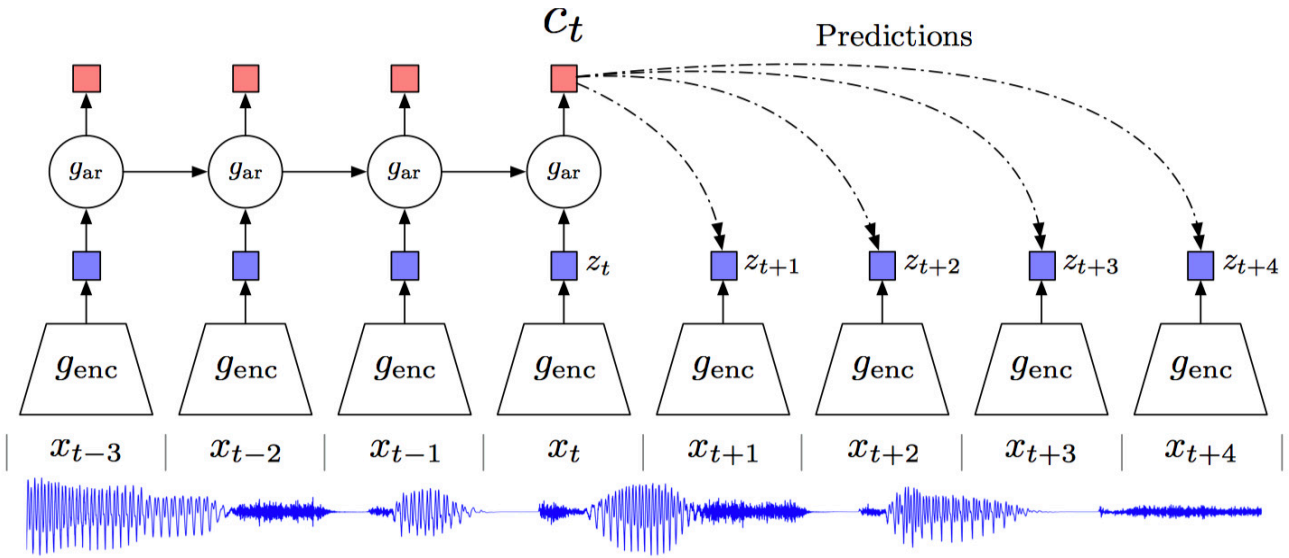

Figure 1: Overview of Contrastive Predictive Coding, the proposed representation learning approach. Although this figure shows audio as input, we use the same setup for images, text and reinforcement learning.

Basic Intuition
Computer Vision

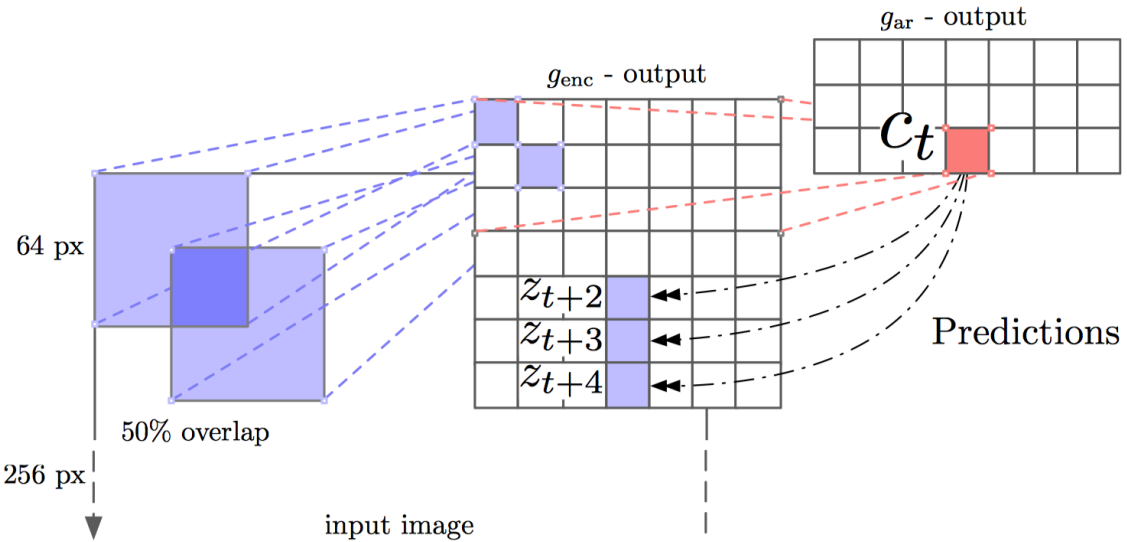

Figure 4: Visualization of Contrastive Predictive Coding for images (2D adaptation of Figure 1).

Modelling $p(x|c)$ is not useful for extracting the shared information

Instead, predict future based on context that maximizes MI

$$I(x;c) = \sum_{x,c} p(x,c) \log \frac{p(x|c)}{p(x)}. \tag{1}$$

$$f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})} \tag{2}$$

$$f_k(x_{t+k}, c_t) = \exp\left(z_{t+k}^T W_k c_t\right), \tag{3}$$

Mathematical View

InfoNCE and Mutual Information

$X = \{x_1, x_2, \ldots, x_N\}$ of which $N - 1$ are negative samples and 1 positive sample.

$$\mathcal{L}_{\mathrm{N}} = -\underset{X}{\mathbb{E}}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right] \tag{4}$$

$$p(d=i|X,c_t) = \frac{p(x_i|c_t)\prod_{l\neq i} p(x_l)}{\sum_{j=1}^{N} p(x_j|c_t)\prod_{l\neq j} p(x_l)}$$
$$= \frac{\frac{p(x_i|c_t)}{p(x_i)}}{\sum_{j=1}^{N} \frac{p(x_j|c_t)}{p(x_j)}}. \quad (5)$$

As we see, the optimal value of $f(x_{t+k}, c_t)$ is proportional to $\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$

How does this loss reflect the density?

$$\mathcal{L}_{\mathrm{N}}^{\mathrm{opt}} = -\mathop{\mathbb{E}}_{X} \log \left[\frac{\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}}{\frac{p(x_{t+k}|c_t)}{p(x_{t+k})} + \sum_{x_j \in X_{\mathrm{neg}}} \frac{p(x_j|c_t)}{p(x_j)}} \right] \tag{6}$$

$$= \mathop{\mathbb{E}}_{X} \log \left[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)} \sum_{x_j \in X_{\mathrm{neg}}} \frac{p(x_j|c_t)}{p(x_j)} \right] \tag{7}$$

$$\approx \mathop{\mathbb{E}}_{X} \log \left[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)} (N-1) \mathop{\mathbb{E}}_{x_j} \frac{p(x_j|c_t)}{p(x_j)} \right] \tag{8}$$

$$= \mathop{\mathbb{E}}_{X} \log \left[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)} (N-1) \right] \tag{9}$$

$$\geq \mathop{\mathbb{E}}_{X} \log \left[\frac{p(x_{t+k})}{p(x_{t+k}|c_t)} N \right] \tag{10}$$

$$= -I(x_{t+k}, c_t) + \log(N), \tag{11}$$

$I(x_{t+k}, c_t) \geq \log(N) - \mathcal{L}_N$, i.e. Decrease $\mathcal{L}_N$, increase lower bound of $I(x_{t+k}, c_t)$

- Encoder --- strided convolutional neural network that runs directly on the 16KHz PCM.
- Five convolutional layers with strides [5, 4, 2, 2, 2], filter-sizes [10, 8, 4, 4, 4] and 512 hidden units with ReLU activations.

- One feature vector = 10 ms of speech

- Autoregressive unit --- GRU RNN with 256-dimensional hidden state.

- The output of the GRU at every timestep is used as the context c from which we predict 12 timesteps in the future using the contrastive loss.

Experimental Setup: Speech

Method	ACC
Phone classification	
Random initialization	27.6
MFCC features	39.7
CPC	64.6
Supervised	74.6
Speaker classification	
Random initialization	1.87
MFCC features	17.6
CPC	97.4
Supervised	98.5

Table 1: LibriSpeech phone and speaker classification results. For phone classification there are 41 possible classes and for speaker classification 251. All models used the same architecture and the same audio input sizes.

Method	ACC
#steps predicted	
2 steps	28.5
4 steps	57.6
8 steps	63.6
12 steps	64.6
16 steps	63.8
Negative samples from	
Mixed speaker	64.6
Same speaker	65.5
Mixed speaker (excl.)	57.3
Same speaker (excl.)	64.6
Current sequence only	65.2

Table 2: LibriSpeech phone classification ablation experiments. More details can be found in Section 3.1.

Experimental Results: Speech

- Input image: 256 X 256, organized as 7 X 7 grid each of the cell is 64 X 64 image patch.
- Encoder ---- ResNet-v2-101 encoder.
- Take the outputs from the third residual block, and spatially mean-pool to get a single 1024-d vector per 64x64 patch. This results in a 7x7x1024 tensor.

- Autoregressive unit ---- PixelCNN-style autoregressive model

- Predict up to five rows from the 7x7 grid,

Experimental Setup: Vision

Method	Top-1 ACC
Using AlexNet conv5	
Video [28]	29.8
Relative Position [11]	30.4
BiGan [35]	34.8
Colorization [10]	35.2
Jigsaw [29] *	38.1
Using ResNet-V2	
Motion Segmentation [36]	27.6
Exemplar [36]	31.5
Relative Position [36]	36.2
Colorization [36]	39.6
CPC	**48.7**

Table 3: ImageNet top-1 unsupervised classification results. *Jigsaw is not directly comparable to the other AlexNet results because of architectural differences.

Method	Top-5 ACC
Motion Segmentation (MS)	48.3
Exemplar (Ex)	53.1
Relative Position (RP)	59.2
Colorization (Col)	62.5
Combination of MS + Ex + RP + Col	69.3
CPC	**73.6**

Table 4: ImageNet top-5 unsupervised classification results. Previous results with MS, Ex, RP and Col were taken from [36] and are the best reported results on this task.

Experimental Results: Vision

Questions?