Information Dropout: Learning Optimal Representations Through Noisy Computation

Alessandro Achille and Stefano Soatto

Presenter: Thang Duong

1. Optimal Representation and the Information Bottleneck loss

a. Define optimal Representation

- **Sufficient**: cross-entropy loss enforce sufficient representation
- **Minimality**: typically reduce the number of dimension. Better to measure with mutual information
- **Invariance**: all features are independent of each other

(My opinion) Why bother learning representation?

$$\min_{\theta \in \mathbb{R}^d} L = \sum_i < x_i, \theta >, \; x_i \in \mathbb{R}^d$$

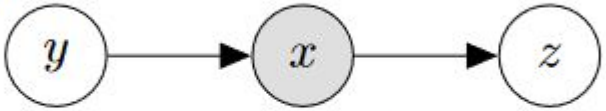
$$<=> \min_{B \in \mathbb{R}^{d \times m}, w \in \mathbb{R}^m} = \sum_i < x_i, Bw >$$

$$= \sum_i < (x_i^\top B)^\top, w >$$

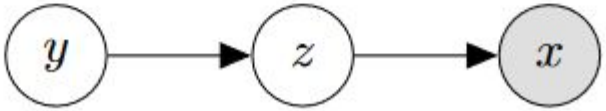
1. Optimal Representation and the Information Bottleneck loss

b. The Information Bottleneck loss

(i) $\mathbf{z}$ is a **representation** of $\mathbf{x}$; that is, its distribution depends only on $\mathbf{x}$, as expressed by the following Markov chain:



(ii) $\mathbf{z}$ is **sufficient** for the task $\mathbf{y}$, that is $I(\mathbf{x};\mathbf{y}) = I(\mathbf{z};\mathbf{y})$, expressed by the Markov chain:



$$\begin{aligned} \text{minimize} \quad & I(\mathbf{x};\mathbf{z}) \\ \text{s.t.} \quad & I(\mathbf{x};\mathbf{y}|\mathbf{z}) = 0. \end{aligned}$$

1. Optimal Representation and the Information Bottleneck loss

b. The Information Bottleneck loss

$$\begin{aligned} &\text{minimize} && I(\mathbf{x};\mathbf{z}) \\ &\text{s.t.} && I(\mathbf{x};\mathbf{y}|\mathbf{z}) = 0. \end{aligned}$$

$$\begin{aligned} \min \mathcal{L}_0 &= I(x, y \mid z) + \beta I(x; z) \\ &= H(y \mid z) - H(y \mid x) + \beta I(x; z) \\ <=> \min \mathcal{L} &= H(y \mid z) + \beta I(x; z) \\ &\approx \mathbb{E}_{x,y\sim p(x,y)} \left[\mathbb{E}_{z\sim p_\theta(z|x)} \left[-\log p_\theta(y \mid z) \right] \right] + \beta KL \left(p_\theta(z \mid x) || p_\theta(z) \right) \\ <=> \min \mathcal{L}_1 &= I(y; z) - \beta_1 I(x; z) \end{aligned}$$

The Information Bottleneck loss shown in the “Deep Variational Information Bottleneck” paper

1. Optimal Representation and the Information Bottleneck loss

c. Disentanglement

d. Loss function

$$\mathrm{TC}(\mathbf{z}) := \mathrm{KL}(q(\mathbf{z}) \parallel \textstyle\prod_j q_j(z_j))$$

The paper points to [*], who shows that a more complex prior (without disentanglement constraint) returns better compression result

Proposition 1. *The minimization problem*

$$\text{minimize}_p \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z\sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \\ + \beta\left\{\mathrm{KL}(p(\mathbf{z}|\mathbf{x}_i) \parallel p(\mathbf{z})) + \mathrm{TC}(\mathbf{z})\right\},$$

Nothing says that p(z|x) is disentangle

is equivalent to the following minimization in two variables

$$\text{minimize}_{p,q} \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z\sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \\ + \beta\,\mathrm{KL}(p(\mathbf{z}|\mathbf{x}_i) \parallel \textstyle\prod_{i=1}^{|z|} q_i(\mathbf{z}_i)).$$

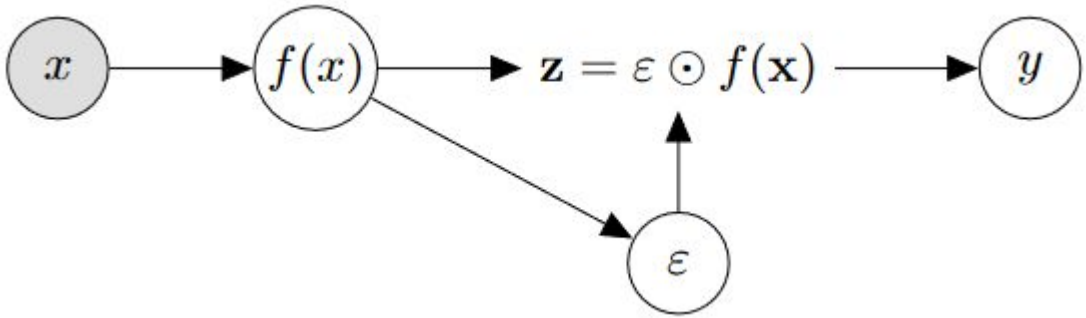
*D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in Advances in Neural Information Processing Systems, 2016, pp. 4743–4751.

2. Information Dropout

Adding stochastic noise: $\varepsilon \sim p_{\alpha(\mathbf{x})}(\varepsilon) = \log \mathcal{N}\left(0, \alpha_\theta^2(\mathbf{x})\right)$

$z = \varepsilon \odot f(x)$

- \odot is element-wise product (Bernoulli distribution => vanilla dropout)
- The choice of log-normal distribution is to simplify the calculation for the KL divergence term with invariant z



2. Information Dropout

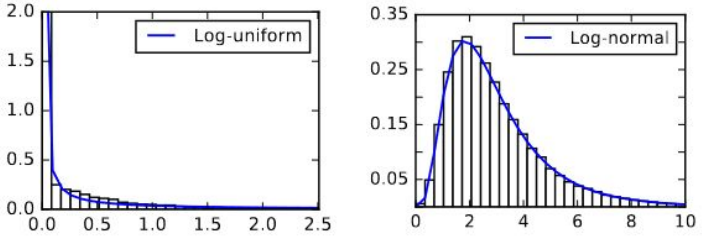
Choosing prior:

- ReUL + Softmax only NNs: scale invariant
- => want scale-invariant prior: $q(\log(z)) = c$ or $q(z) = c/z$
- Because of RELU, assume

$$q(z = 0) = q_0, 0 \leq q_0 \leq 1$$

- Final: $q(z) = q_0 \delta_0(z) + c/z$

Sub-gradient ?



(a) Histogram of ReLU activations (b) Histogram of Softplus activations

Fig. 1: Comparison of the empirical distribution $p(z)$ of the post-noise activations with our proposed prior when using: (a) ReLU activations, for which we propose a log-uniform prior, and (b) Softplus activations, for which we propose a log-normal prior. In both cases, the empirical distribution approximately follows the proposed prior. Both histograms where obtained from the last dropout layer of the All-CNN-32 network described in Table 2, trained on CIFAR-10.

2. Information Dropout

Calculating the KL divergent with the new prior

Proposition 2 (Information dropout cost for ReLU). *Let* $z = \varepsilon \cdot f(x)$, *where* $\varepsilon \sim p_\alpha(\varepsilon)$, *and assume* $p(z) = q\delta_0(z) + c/z$. *Then, assuming* $f(x) \neq 0$, *we have*

$$\mathrm{KL}(p_\theta(z|x) \,\|\, p(z)) = -H(p_{\alpha(x)}(\log \varepsilon)) + \log c$$

In particular, if $p_\alpha(\varepsilon)$ *is chosen to be the log-normal distribution* $p_\alpha(\varepsilon) = \log \mathcal{N}(0, \alpha_\theta^2(x))$, *we have*

$$\mathrm{KL}(p_\theta(z|x) \,\|\, p(z)) = -\log \alpha_\theta(x) + const. \tag{5}$$

If instead $f(x) = 0$, *we have*

$$\mathrm{KL}(p_\theta(z|x) \,\|\, p(z)) = -\log q.$$

Proposition 3 (Information dropout cost for Softplus). *Let* $z = \varepsilon \cdot f(x)$, *where* $\varepsilon \sim p_\alpha(\varepsilon) = \log \mathcal{N}(0, \alpha_\theta^2(x))$, *and assume* $p_\theta(z) = \log \mathcal{N}(\mu, \sigma^2)$. *Then, we have*

$$\mathrm{KL}(p_\theta(z|x) \,\|\, p(z)) = \frac{1}{2\sigma^2}\left(\alpha^2(x) + \mu^2\right) - \log \frac{\alpha(x)}{\sigma} - \frac{1}{2}. \tag{6}$$

3. Connection to other frameworks

- This paper: $\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{z \sim p_\theta(z|x_i)} \left[-\log p_\theta(y_i \mid z) \right] + \beta KL \left(p_\theta(z \mid x_i) || \Pi_j q_\theta(z_j) \right)$
- VAE: $\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{z \sim p_\theta(z|x_i)} \left[-\log p_\theta(x_i \mid z) \right] + KL \left(p_\theta(z \mid x_i) || \Pi_j q_\theta(z_j) \right)$
- References on different prior p(z) and different value β
- Independent Component Analysis (ICA):

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{z \sim p_\theta(z|x_i)} \left[-\log p_\theta(x_i \mid z) \right] + \gamma TC(z)$$

- $\varepsilon \sim p_{\alpha(x)}(\varepsilon)$ as Bernoulli distribution results in vanilla Dropout

4. Experiments

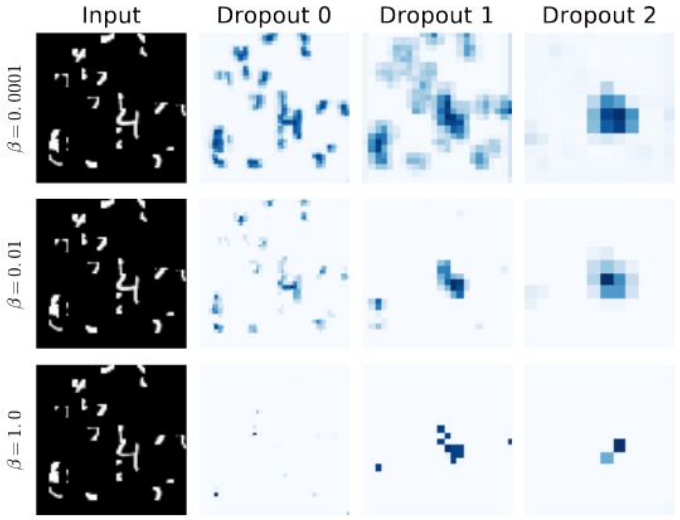
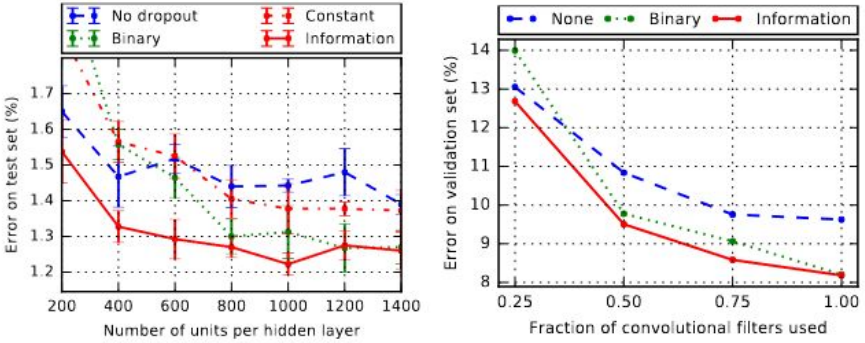


Fig. 2: Plot of the total KL-divergence at each spatial location in the first three Information Dropout layers (of sizes 48x48, 24x24 and 12x12 respectively) of All-CNN-96 (see Table 2) trained on Cluttered MNIST with different values of β. This measures how much information from each part of the image the Information Dropout layer is transmitting to the next layer. For small β information about the nuisances is transmitted to the next layers, while for higher values of β the dropout layers drop the information as soon as the receptive field is big enough to recognize it as a nuisance. The resulting representation is thus more robust to nuisances, improving generalization. Notice that the noise added by Information Dropout is tailored to the specific sample, to the point that the digit can be localized from the noise mask.

4. Experiments

Fixed rate drop-out (Constant) is worse than adaptive (Information)



(a) MNIST (b) CIFAR-10

Fig. 3: (a) Average classification error on MNIST over 3 runs of several dropout methods applied to a fully connected network with three hidden layers and ReLU activations. Information dropout outperforms binary dropout, especially on smaller networks, possibly because dropout severely reduces the already limited capacity of the network, while Information Dropout can adapt the amount of noise to the data and the size of the network. Information dropout also outperforms a dropout layer that uses constant log-normal noise with the same variance, confirming the benefits of adaptive noise. (b) Classification error on CIFAR-10 for several dropout methods applied to the All-CNN-32 network (see Table 2) using Softplus activations.

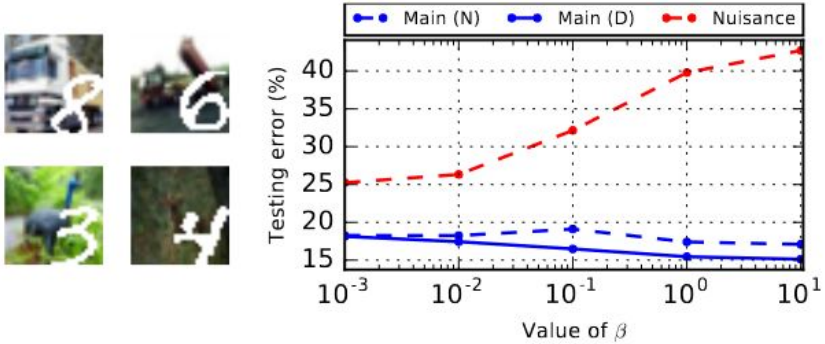


Fig. 4: A few samples from our Occluded CIFAR dataset and the plot of the testing error on the main task (classifying the CIFAR image) and on the nuisance task (classifying the occluding MNIST digit) as β varies. For both tasks, we use the same representation of the data trained for the main task using Information Dropout. For larger values of β the representation is increasingly more invariant to nuisances, making the nuisance classification task harder, but improving the performance on the main task by preventing overfitting. For the nuisance task, we test using the learned noisy representation of the data, since we are interested specifically in the effects of the noise. For the main task, we show the result both using the noisy representation (N), and the deterministic representation (D) obtained by disabling the noise at testing time.

4. Experiments

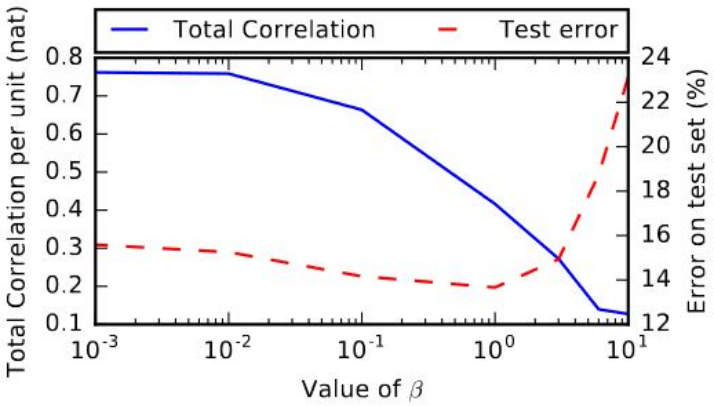


Fig. 5: For different values of β, plot of the test error and total correlation of the final layer of the All-CNN-32 network with Softplus activations trained on CIFAR-10 with 25% of the filters. Increasing β the test error decreases (we prevent overfitting) and the representation becomes increasingly disentangled. When β is too large, it prevents information from passing through, jeopardizing sufficiency and causingi a drastic increase in error.

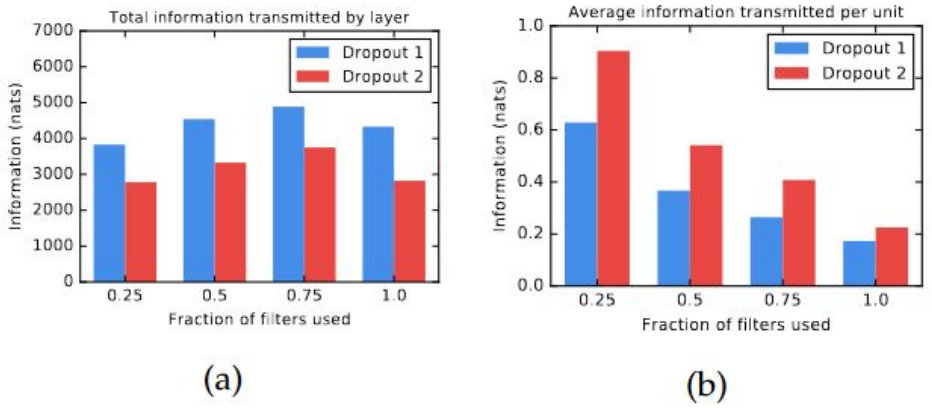


Fig. 6: Plots of (a) the total information transmitted through the two dropout layers of a All-CNN-32 network with Softplus activations trained on CIFAR and (b) the average quantity of information transmitted through each unit in the two layers. From (a) we see that the total quantity of information transmitted does not vary much with the number of filters and that, as expected, the second layer transmits less information than the first layer, since prior to it more nuisances have been disentangled and discarded. In (b) we see that when we decrease the number of filters, we force each single unit to let more information flow (i.e. we apply less noise), and that the units in the top dropout layer contain on average more information relevant to the task than the units in the bottom dropout layer.