Parametric Information Bottleneck [1]

SAIL@UNIST Group Meeting

T.T. Nguyen and J. Choi

SAIL@UNIST

November 3, 2018

[1] v4: 18/11/02

Contents

Contents

Statistical Artificial Intelligence Laboratory @UNIST

Introduction

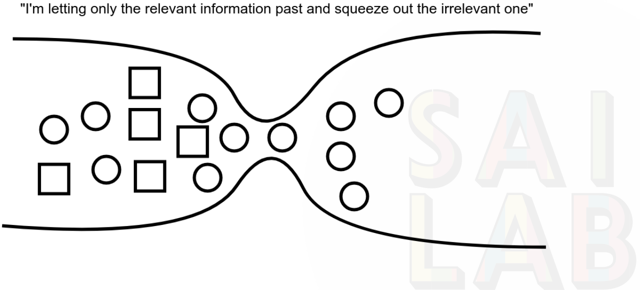
- Deep neural networks (DNNs) = flexible modeling capability
  - Multi-layered neural networks
- Learning principle = a principled way of exploiting a model for certain tasks
- The Maximum Likelihood Estimate (MLE) principle:
  - A de-facto learning principle for DNNs
  - Maximizing the model's likelihood of seeing the training data

MLE + DNNs = ?

- MLE: ignores the special topology of DNNs during the learning
  - generic to many models, not dedicatedly tailored for DNNs
  - MLE sees the entire neural architecture $f(\boldsymbol{x})$ as a whole, without taking advantage of the hierarchical structure of the neural function f
- We need a clever way to exploit the topology of DNNs for learning!
  $\rightarrow$ Parametric Information Bottleneck (PIB)!

Parametric Information Bottleneck: A quick look

Informally, PIB principle is assuring that <u>each</u> layer of DNNs maximally preserves the information relevant to the task while it is being compressed!



PIB = an efficient way to induce layer-wise relevance-compression trade-offs to DNNs!

More about PIB

Though being conceptually simple, the layer-wise compression-relevance trade-offs pose challenges theoretically and empirically:

1. Is it possible to obtain the optimal compression-relevance trade-offs simultaneously at all layers?
2. Computing compression and relevance in DNNs is highly intractable

The main contributions in PIB is efficiently inducing layer-wise compression-relevance trade-offs to DNNs by addressing 1. and proposing an approximate mutual information to 2.

Contents

Contents

Information Bottleneck Principle: Problem Setting

Learning problem setting: $X =$ input random variable (RV), $Y =$ output random variable

- E.g., supervised learning: $X =$ input images, $Y =$ image label
- E.g., unsupervised learning: $X =$ input image, $Y = X =$ reconstruction image

Information Bottleneck Principle

- A principled way of extracting relevant information in one variable, X about another variable, Y.
- Encode X into an intermediate representation,

$$X \xrightarrow{p(z|x)} Z,$$

in such a way that Z preserves as much of **relevant information** about Y as possible.

Information Bottleneck Principle

IB = optimizing the compression-relevance trade-off (the following problems are equivalent):

- Compression-Relevance Function [Tishby et al., 1999, Slonim and Weiss, 2002]:

$$\underset{P(Z|X): I(Z;Y) \geq D}{\arg\min} \quad I(Z;X)$$

  where D is a positive number specifying the minimum relevance.

- Lagrangian multiplier [Tishby et al., 1999]:

$$\underset{P(Z|X)}{\arg\min}\, I(Z;X) - \beta I(Z;Y)$$

  where β is a (positive) Lagrangian multiplier.

Information Bottleneck Principle

- **Compression** (measure) = representation complexity = $I(Z;X)$
- **Relevance** = predictive power = $I(Z;Y)$
- Mutual Information:

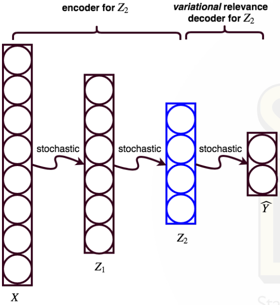
$$I(Z;Y) = \int p(\boldsymbol{z},\boldsymbol{y}) \log \frac{p(\boldsymbol{z},\boldsymbol{y})}{p(\boldsymbol{z})p(\boldsymbol{y})} d\boldsymbol{z} d\boldsymbol{y},$$

  Intuitively, $I(Z;Y)$ = the amount of information that Z contains about Y.
- Compression-relevance trade-off:
  - You might lose relevant information as well if you compress (the representation) too much!
  - But if you allow too much information (in the representation), you might let past irrelevant information (which is not useful for the task performance)

Contents

Parametric Information Bottleneck

- PIB = Layer-wise application of IB to DNNs
- Topology of DNNs = Markov chain structure: $Y \rightarrow X \rightarrow Z_l \rightarrow Z_{l+1} \rightarrow \hat{Y}$



- Encoder $P(Z_l|X)$ is induced by the network architecture and Bayes' Rule, e.g., $P(Z_1|X) = \sigma(W_1 X + b_1)$ in sigmoid-activated fully-connected neural network

PIB

PIB = layer-wise multi-objective Information Bottleneck:

$$\forall 1 \leq l \leq L, \min_{P(Z_l|X)} \mathcal{L}_l[P(Z_l|X)] \tag{1}$$
$$\text{where } \mathcal{L}_l[P(Z_l|X)] := I(Z_l; X) - \beta_l I(Z_l; Y)$$

PIB

Unfortunately, we cannot achieve the layer-wise optimality for simultaneously all layers [2]

Theorem 3.1 (Conflicting Information Optimality). *Given* $Y \to X \to Z_1 \to Z_2, Z_2 \not\to Z_1, \beta_1 > 0$, *and* $\beta_2 > 0$, *then* $\mathcal{L}_1$ *and* $\mathcal{L}_2$ *defined by the layer-wise multi-objective Information Bottleneck are conflicting, i.e., there does not exist a single solution that minimizes* $\mathcal{L}_1$ *and* $\mathcal{L}_2$ *simultaneously.*

$\rightarrow$ The layers to compromise their information optimality $\rightarrow$ we propose two simple compromised strategies: *JointPIB* and *GreedyPIB*

[2] Detailed proof at `https://arxiv.org/abs/1712.01272`

PIB: comprimised optimality

- JointPIB:

$$\mathcal{L}^{joint} := \sum_{l=0}^{L} \gamma_l \tilde{\mathcal{L}}_l$$

- GreedyPIB: applies PIB progressively in a greedy manner. In other words, *GreedyPIB* tries to obtain the conditional optimality of a current layer which is conditioned on the achieved conditional optimality of the previous layers.

Intractability of Mutual Information

- Compression and Relevance (which are mutual information by nature) of high-dimensional random variables in PIB are highly intractable
- We propose *Variational Relevance* and *Variational Compression* to address that issue for DNNs

Relevance in PIB

- Relevance:

$$I(Z_l; Y) = H(Y) - H(Y|Z_l)$$
$$H(Y|Z_l) = -\int p(\boldsymbol{y}, \boldsymbol{z}_l) \log p(\boldsymbol{y}|\boldsymbol{z}_l) d\boldsymbol{y} d\boldsymbol{z}_l$$

- Relevance decoder:

$$p(\boldsymbol{y}|\boldsymbol{z}_l) = \int p_D(\boldsymbol{x}, \boldsymbol{y}) \frac{p(\boldsymbol{z}_l|\boldsymbol{x})}{p(\boldsymbol{z}_l)} d\boldsymbol{x}$$

Variational Relevance

- It follows from Jensen's inequality that:

$$\begin{aligned}
H(Y|Z_l) &= -\int p(\boldsymbol{y}|\boldsymbol{z}_l)p(\boldsymbol{z}_l)\log \overset{\text{relevance decoder}}{\overset{\downarrow}{\boxed{p(\boldsymbol{y}|\boldsymbol{z}_l)}}} d\boldsymbol{y}\, d\boldsymbol{z}_l \\
&\leq -\int p(\boldsymbol{y}|\boldsymbol{z}_l)p(\boldsymbol{z}_l)\log \overset{\textit{variational}\text{ relevance decoder}}{\overset{\downarrow}{\boxed{q(\boldsymbol{y}|\boldsymbol{z}_l)}}} d\boldsymbol{y}\, d\boldsymbol{z}_l \\
&= -\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})_D}\mathbb{E}_{\boldsymbol{z}_l|\boldsymbol{x}}\log \overset{\textit{variational}\text{ relevance decoder}}{\overset{\downarrow}{\boxed{q(\boldsymbol{y}|\boldsymbol{z}_l)}}} =: \tilde{H}(Y|Z_l)
\end{aligned} \tag{2}$$

  where $q(\boldsymbol{y}|\boldsymbol{z}_l)$ is any probability distribution.
- In PIB, we set $q(Y|Z_l) = P(\hat{Y}|Z_l)$: re-use the the higher-level network architecture to define variational relevance decoder
- $\tilde{H}(\hat{Y}|Z_l) =:$ variational conditional relevance (VCR)

Variational Relevance

Variational Conditional Relevance (VCR) generalizes MLE to all layers in DNNs!

Theorem 3.2 (Information on the extreme layers). *The VCR of the lowest-level (so-called **super**) layer (i.e., $l = 0$) is the negative log-likelihood (NLL) function of the neural network, i.e.,*

$$\tilde{H}(Y|Z_0) = -\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})_D}\left[\log p(\hat{\boldsymbol{y}}|\boldsymbol{x})\right]. \tag{18}$$

*Similarly, the VCR of the highest-level layer (i.e., $l = L$) equals that of the **compositional** layer $Z = (Z_1, Z_2, ..., Z_L)$, a composite of all hidden layers; in addition, their VCR is an upper bound on the NLL:*

$$\tilde{H}(Y|Z_L) = \tilde{H}(Y|Z) \geq -\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})_D}\left[\log p(\hat{\boldsymbol{y}}|\boldsymbol{x})\right]. \tag{19}$$

Variational Compression

- Avoid directly estimating $I(Z_l; X)$ by instead resorting to its upper bound $I(Z_l; Z_{l-1})$
- Approximate $I(Z_l; Z_{l-1})$ using a mean-field (factorized) variational distribution $r(\boldsymbol{z}_l) = \prod_{i=1}^{n_l} r(z_{l,i})$:

$$
\begin{aligned}
I(Z_l; X) &\leq I(Z_l; Z_{l-1}) \\
&= \int p(\boldsymbol{z}_l|\boldsymbol{z}_{l-1}) p(\boldsymbol{z}_{l-1}) \log \frac{p(\boldsymbol{z}_l|\boldsymbol{z}_{l-1})}{p(\boldsymbol{z}_l)} d\boldsymbol{z}_l d\boldsymbol{z}_{l-1} \\
&\leq \int p(\boldsymbol{z}_l|\boldsymbol{z}_{l-1}) p(\boldsymbol{z}_{l-1}) \log \frac{p(\boldsymbol{z}_l|\boldsymbol{z}_{l-1})}{r(\boldsymbol{z}_l)} d\boldsymbol{z}_l d\boldsymbol{z}_{l-1} \\
&= \mathbb{E}_{\boldsymbol{z}_{l-1}} \sum_{i=1}^{n_l} D_{KL}\left[p(z_{l,i}|\boldsymbol{z}_{l-1}) || r(z_{l,i})\right] \\
&=: \tilde{I}(Z_l; Z_{l-1})
\end{aligned} \tag{3}
$$

Contents

Classification and Adversarial Robustness

- PIB offers a DNN model with competitive classification performance and robustness against adversarial attacks

Model	Classification MNIST (Error %)	Classification CIFAR10 (Accuracy %)	Adv. Robustness (%) Targeted	Adv. Robustness (%) Untargeted
DET	1.73	53.91	00.00	00.00
VIB Alemi et al. (2017)	1.45	54.41	83.70	93.10
SFNN Raiko et al. (2015)	1.44	55.94	83.00	95.20
GreedyPIB	1.54	**57.61**	83.21	94.30
JointPIB	**1.36**	55.62	**84.16**	**96.00**

Table 1: The performance of PIB for classification and adversarial robustness on MNIST and CIFAR10 in comparison with MLE and a partially information-theoretic treatment VIB.

Learning dynamics

- Closer look at the inside of learning: PIB preserves relevant information throughout all layers during the learning

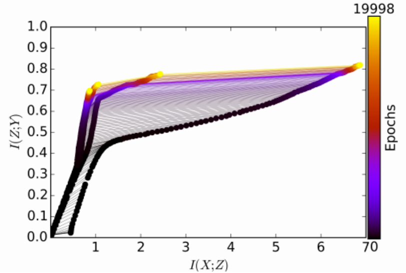


Figure 1. Learning dynamics of SFNN

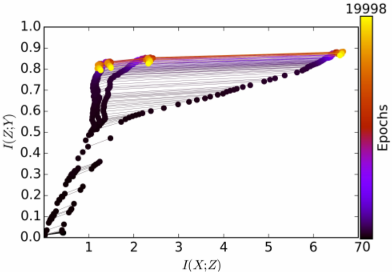


Figure 2. Learning dynamics of JointPIB

Learning dynamics

- GreedyPIB progressively preserves relevant information throughout all layers in a greedy manner.

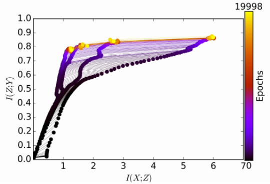


Figure 3. Learning dynamics of GreedyPIB at $l = 1$.

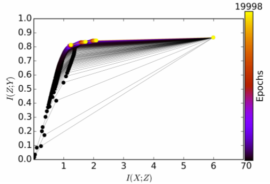


Figure 4. Learning dynamics of GreedyPIB at $l = 2$.

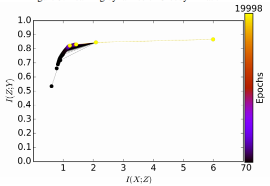


Figure 5. Learning dynamics of GreedyPIB at $l = 3$.

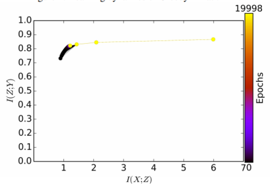


Figure 6. Learning dynamics of GreedyPIB at $l = 4$.

Thank you for listening!

Statistical Artificial Intelligence Laboratory @UNIST

References

Slonim, N. and Weiss, Y. (2002).
Maximum likelihood and the information bottleneck.
In *Advances in Neural Information Processing Systems 15 [Neural Information Processing Systems, NIPS 2002, December 9-14, 2002, Vancouver, British Columbia, Canada]*, pages 335–342.

Tishby, N., Pereira, F. C., and Bialek, W. (1999).
The information bottleneck method.
In Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing.