

Information Bottleneck View of Deep Learning

Shihua Zhang

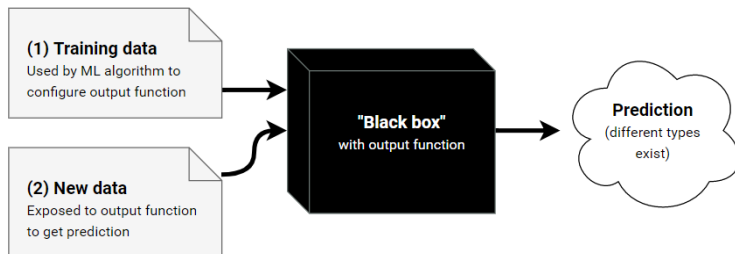
November 21, 2024

- 1 Background
- 2 Information Bottleneck for Relevance
- 3 Information Bottleneck Views of DNNs
- 4 Information Bottleneck as Optimization Objective
- 5 Summary

- 1 Background
- 2 Information Bottleneck for Relevance
- 3 Information Bottleneck Views of DNNs
- 4 Information Bottleneck as Optimization Objective
- 5 Summary

What does Deep Learning Do?

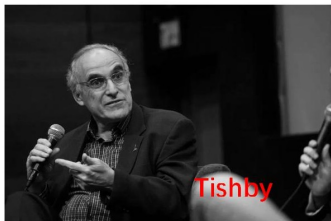
- Deep learning achieves lots of successes in diverse fields.



- How to **open the black-box** on its success?
- Some basic problems have not been solved:
 - Why stochastic gradient descent (SGD) works?
 - What's happening over training with SGD?
 - How sample size affects the training and generalization?
 - What's the optimal solutions of DNNs when they converge?

How to Open the Black-box?

- Tishby *et al.* [10]: open the black-box via **information theory** by
 - quantify the **information flow** along layers of DNNs.
 - characterize the **phases** over training.



Hinton: “It’s extremely interesting. I have to listen to it another 10,000 times to really understand it”.

Outline

- 1 Background
- 2 Information Bottleneck for Relevance**
- 3 Information Bottleneck Views of DNNs
- 4 Information Bottleneck as Optimization Objective
- 5 Summary

Sufficient Statistic

What captures the **relevant** properties in samples about a parameter?

- Given i.i.d. samples $x^{(n)} \sim p(x|\theta)$

Sufficient Statistic

What captures the **relevant** properties in samples about a parameter?

- Given i.i.d. samples $x^{(n)} \sim p(x|\theta)$

Definition (Sufficient Statistics)

A *sufficient statistic*: $T(x^{(n)})$ is a function of the sample such that

$$p\left(x^{(n)} \mid T\left(x^{(n)}\right) = t, \theta\right) = p\left(x^{(n)} \mid T\left(x^{(n)}\right) = t\right) \quad (1)$$

- Sufficient statistics contain everything about θ from samples $x^{(n)}$.

Sufficient Statistic

What captures the **relevant** properties in samples about a parameter?

- Given i.i.d. samples $x^{(n)} \sim p(x|\theta)$

Definition (Sufficient Statistics)

A *sufficient statistic*: $T(x^{(n)})$ is a function of the sample such that

$$p\left(x^{(n)} \mid T\left(x^{(n)}\right) = t, \theta\right) = p\left(x^{(n)} \mid T\left(x^{(n)}\right) = t\right) \quad (1)$$

- Sufficient statistics contain everything about θ from samples $x^{(n)}$.

There are always trivial sufficient statistics - e.g., the sample itself.

Definition (Minimal Sufficient Statistics)

One sufficient statistic $S(x^n)$ is called a *minimal sufficient statistic (MSS)* for θ in $p(x|\theta)$ if it is a function of any other sufficient statistics $T(x^{(n)})$.

- $S(x^{(n)})$ gives the coarser sufficient description of the samples.
- S is unique (up to 1-1 map).

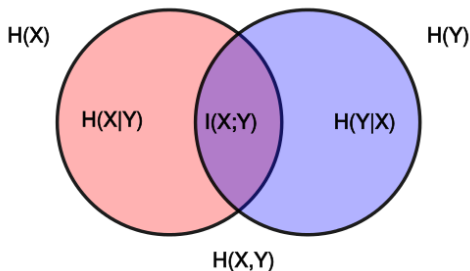
Sufficiency and Information

Definition (Mutual Information) [4]

For any two random variables X and Y with joint pdf $P(X = x, Y = y) = p(x, y)$, Shannon's mutual information $I(X; Y)$ is defined as

$$I(X; Y) = \mathbb{E}_{p(x,y)} \log \frac{p(x, y)}{p(x)p(y)} = D_{KL}(p(x, y) \| p(x)p(y)) \quad (2)$$

- $I(X; Y) = H(X) - H(X | Y) = H(Y) - H(Y | X) \geq 0$



Properties of Mutual Information

Key properties of mutual information:

Theorem (Data-processing inequality)

When $X \rightarrow Y \rightarrow Z$ form a Markov chain, then

$$I(X; Z) \leq I(X; Y)$$

— data processing cannot increase (mutual) information

Warm-up: prove the data processing inequality

Hint: chain rule of mutual information

Warm-up: prove the data processing inequality

Hint: chain rule of mutual information

Chain rule

$$\begin{aligned} I(X; Y, Z) &= I(X; Z) + I(X; Y|Z) \\ &= I(X; Y) + I(X; Z|Y) \end{aligned}$$

Warm-up: prove the data processing inequality

Hint: chain rule of mutual information

Chain rule

$$\begin{aligned} I(X; Y, Z) &= I(X; Z) + I(X; Y|Z) \\ &= I(X; Y) + \cancel{I(X; Z|Y)} \end{aligned}$$

According to the property of Markov chain, $I(X; Z|Y) = 0$, we obtain the DPI immediately.

Sufficiency and Information

We can characterize sufficiency and minimality using mutual information:

Theorem (Sufficiency and Information)

- T is sufficient statistics for θ in $p(x | \theta) \iff$

$$I\left(T\left(x^{(n)}\right); \theta\right)=I\left(x^{(n)} ; \theta\right)$$

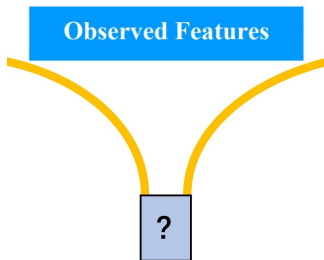
- If S is minimal sufficient statistics for θ in $p(x | \theta)$, then:

$$I\left(S\left(x^{(n)}\right) ; x^{(n)}\right) \leq I\left(T\left(x^{(n)}\right) ; x^{(n)}\right)$$

That is, among all sufficient statistics, **minimality** maintains the least mutual information on the samples $x^{(n)}$.

Relation to Learning Theory

- In a supervised manner, only samples $\{x^n, y^n\}_{n=1}^N \sim p(x, y)$ are given.
- Take the samples as random variables (X, Y)
- How to extract an efficient representation of the **relevant information** contained in a large set of features (X) ?
- What information is **relevant**?
- The Information Bottleneck method [11, 12] answers this.



Information Bottleneck: Approximate MSS

- Given $(X, Y) \sim p(x, y)$, it suggests that the learning objective is to find the **relevant** part X with respect to Y , i.e., the MSS,

$$\begin{aligned} \min_T & I(X; T) \\ \text{s.t. } & I(T; Y) = I(X; Y) \end{aligned} \tag{3}$$

- However, $p(x, y)$ is **not known** and only samples are provided.
- $I(X; Y)$ is **intractable**.

Information Bottleneck: Approximate MSS

- Given $(X, Y) \sim p(x, y)$, it suggests that the learning objective is to find the **relevant** part X with respect to Y , i.e., the MSS,

$$\begin{aligned} \min_T & I(X; T) \\ \text{s.t. } & I(T; Y) = I(X; Y) \end{aligned} \quad (3)$$

- However, $p(x, y)$ is **not known** and only samples are provided.
- $I(X; Y)$ is **intractable**.
- Minimal Sufficient Statistics (MSS) can be approximated:

$$\begin{aligned} \min_T & I(X; T) \\ \text{s.t. } & I(T; Y) = \alpha \end{aligned} \quad (4)$$

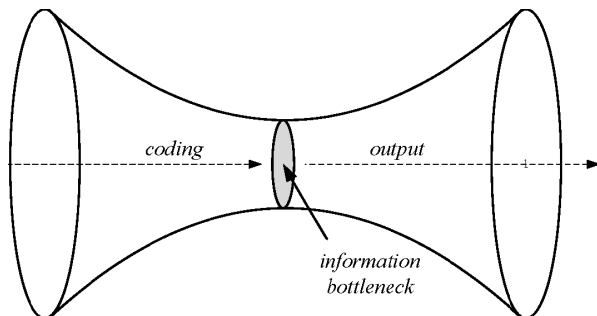
- T is called the **Information Bottleneck** between X and Y .

Why Information Bottleneck?

Definition

Compression: reduction in $I(X; T)$ over the course of training.

The coding is compressed smaller than α , as like the data through the bottleneck.

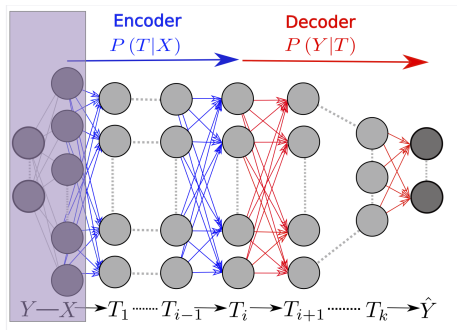


Outline

- 1 Background
- 2 Information Bottleneck for Relevance
- 3 Information Bottleneck Views of DNNs**
- 4 Information Bottleneck as Optimization Objective
- 5 Summary

Information Flow along DNNs

Consider a feed-forward neural network

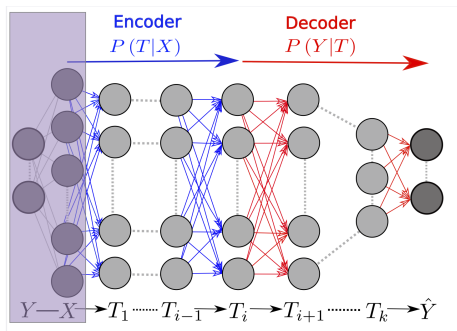


- Take the whole layer as a single random variable, in which the i -th hidden layer representation is processed from X :

$$T_i = \sigma_i (\mathbf{W}_i \sigma_{i-1} (\mathbf{W}_{i-1} \cdots \sigma_1 (\mathbf{W}_1 \mathbf{x}) \cdots)) \quad (5)$$

- Network layers form a Markov chain.

Information Flow along DNNs



- From **data processing inequality**, the information is lost along layers.

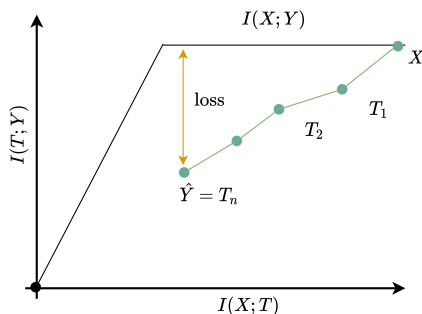
$$I(Y; X) \geq I(Y; T_1) \geq I(Y; T_i) \geq \dots \geq I(Y; \tilde{Y})$$

$$H(X) \geq I(X; T_1) \geq I(X; T_i) \geq \dots \geq I(X; \tilde{Y})$$

Learning by Forgetting?

Looking at networks in the **information plane**:

- **Stacking multiple layers** makes the representation increasingly minimal.
- **Minimizing usual Cross-Entropy loss $H(T, Y)$** maximizes the mutual information $I(T; Y)$.
- **Layer-processing and regularization** of DNNs: maximizing $I(T; Y)$ while minimizing $I(X; T)$ to some extent.



Compression Helps?

Definition

Compression: reduction in $I(X; T)$ over the course of training.

Compression Helps?

Definition

Compression: reduction in $I(X; T)$ over the course of training.

Question 1: Compression helps?

- Compression promotes better generalization [3]

$$P[|\text{err}_{\text{test}} - \text{err}_{\text{train}}| > \epsilon] < O\left(\frac{I(X; T)}{n\epsilon^2}\right) \quad (6)$$

Conjecture: SGD Has Two Phases

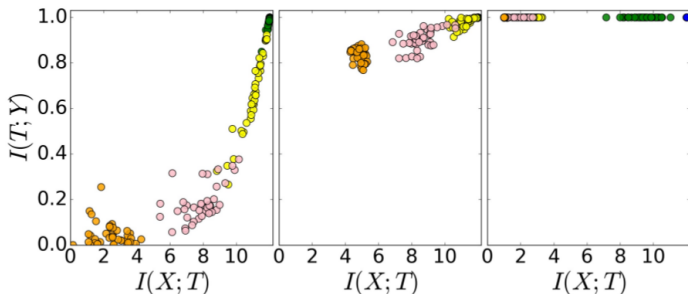


Figure: The **dynamics of mutual information** between hidden layers (red points represent deep layers) and the inputs X or labels Y (Training Start, Middle, End).

- SGD training present two phases:
 - fitting (left \rightarrow middle): increase of $I(X; T)$ and $I(T; Y)$.
 - compression (middle \rightarrow right): increase of $I(T; Y)$ and decrease of $I(X; T)$.

Different Sample Sizes

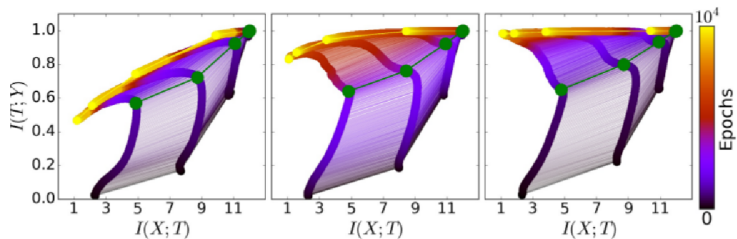


Figure: Training with 5%, 45%, 85% of the data, respectively.

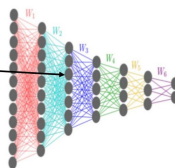
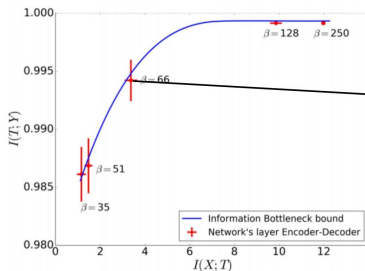
- Small sample size **loses label information** (the compression phase).
- Causes over-fitting, which can be prevented by early stopping.

Converged Layers are Close to the IB Bound

- Given data $\{x^n, y^n\}_{n=1}^N \sim p(x, y)$, **Information Bottleneck (IB) Lagrangian** can be defined as

$$\min_T I(X; T) - \beta I(T; Y) \quad (7)$$

- Given different β s, the IB Lagrangian can be solved.
- The solutions form the IB bound curve (**blue**).
- Claim:** The solutions of DNNs converge to the IB bound: **DNNs are minimizing the IB Lagrangian objective.**



Black Box is Opened? Problem Solved?

However, the findings in [8] shows that these claims may not hold true in the general case ...

- **Compression dynamics** may be a general feature of DNNs.
 - **or may not**, but influenced by the non-linearities employed by DNN.
- Generalization performance may not relate to the **information plane** behaviour
 - **compression may occur** to a subset of features if the task demands it.

Black Box is Opened? Problem Solved?

However, the findings in [8] shows that these claims may not hold true in the general case ...

- **Compression dynamics** may be a general feature of DNNs.
 - **or may not**, but influenced by the non-linearities employed by DNN.
- Generalization performance may not relate to the **information plane** behaviour
 - **compression may occur** to a subset of features if the task demands it.

How to improve the **compression**

- 1 Background
- 2 Information Bottleneck for Relevance
- 3 Information Bottleneck Views of DNNs
- 4 Information Bottleneck as Optimization Objective**
 - Blahut-Arimoto Algorithm for Known Joint Distribution
 - Variational Information Bottleneck
- 5 Summary

Outline

- 1 Background
- 2 Information Bottleneck for Relevance
- 3 Information Bottleneck Views of DNNs
- 4 Information Bottleneck as Optimization Objective**
 - Blahut-Arimoto Algorithm for Known Joint Distribution
 - Variational Information Bottleneck
- 5 Summary

Compression Helps?

Definition

Compression: reduction in $I(X; T)$ over the course of training.

Compression Helps?

Definition

Compression: reduction in $I(X; T)$ over the course of training.

Question 1: Compression helps?

- Compression promotes better generalization [3]

$$P[|\text{err}_{\text{test}} - \text{err}_{\text{train}}| > \epsilon] < O\left(\frac{I(X; T)}{n\epsilon^2}\right) \quad (8)$$

Compression Helps?

Definition

Compression: reduction in $I(X; T)$ over the course of training.

Question 1: Compression helps?

- Compression promotes better generalization [3]

$$P[|\text{err}_{\text{test}} - \text{err}_{\text{train}}| > \epsilon] < O\left(\frac{I(X; T)}{n\epsilon^2}\right) \quad (8)$$

Question 2: How to further compress the representation?

- Solve IB objective with smaller α .

$$\begin{aligned} \min_T & I(X; T) \\ \text{s.t. } & I(T; Y) = \alpha \end{aligned} \quad (9)$$

Information Bottleneck Lagrangian

- The constrained IB objective is equivalent to the following IB Lagrangian,

$$\min_{p(t|x)} I(X; T) - \beta I(T; Y) \quad (10)$$

- $p(t|x)$ is the conditional probability given x (encoder function)
- β is the trade-off hyper-parameter **balancing fitting and compression**.
- **Solvable** when $p(x, y)$ are known [11].

Theorem (Tishby, 1998)

The optimal assignment, that minimizes IB Lagrangian, satisfies the equation

$$p(t | x) = \frac{p(t)}{Z(x, \beta)} \exp \left[-\beta \sum_y p(y | x) \log \frac{p(y | x)}{p(y | t)} \right] \quad (11)$$

where the distribution $p(y | t)$ in the exponent is given via Bayes' rule, as,

$$p(y | t) = \frac{1}{p(t)} \sum_x p(y | x) p(t | x) p(x)$$

Proof Sketch (pt.1)

Introducing Lagrangian multipliers, $\lambda(x)$ for the normalization of the conditional distributions $p(t | x)$ at each x , IB Lagrangian becomes

$$\begin{aligned}\mathcal{L} &= I(X, T) - \beta I(T, Y) - \sum_{x,t} \lambda(x) p(t | x) \\ &= \sum_{x,t} p(t | x) p(x) \log \left[\frac{p(t | x)}{p(t)} \right] - \beta \sum_{t,y} p(t, y) \log \left[\frac{p(t | y)}{p(t)} \right] \\ &\quad - \sum_{x,t} \lambda(x) p(t | x)\end{aligned}\tag{12}$$

Taking derivatives with respect to $p(t | x)$ for given x and t , one obtains

$$\begin{aligned}\frac{\delta \mathcal{L}}{\delta p(t | x)} &= p(x) [1 + \log p(t | x)] - \frac{\delta p(t)}{\delta p(t | x)} [1 + \log p(t)] \\ &\quad - \beta \sum_y \frac{\delta p(t | y)}{\delta p(t | x)} p(y) [1 + \log p(t | y)] \\ &\quad - \beta \frac{\delta p(t)}{\delta p(t | x)} [1 + \log p(t)] - \lambda(x)\end{aligned}$$

Proof Sketch (pt. 2)

According the following derivatives $\frac{\delta p(t)}{\delta p(t|x)} = p(x)$ and $\frac{\delta p(t|y)}{\delta p(t|x)} = p(x | y)$, we have

$$\frac{\delta \mathcal{L}}{\delta p(t|x)} = p(x) \left\{ \log \left[\frac{p(t|x)}{p(t)} \right] - \beta \sum_y p(y|x) \log \left[\frac{p(y|t)}{p(y)} \right] - \frac{\lambda(x)}{p(x)} \right\}$$

Finally, obtain the variational condition:

$$\frac{\delta \mathcal{L}}{\delta p(t|x)} = p(x) \left[\log \frac{p(t|x)}{p(t)} + \beta \sum_y p(y|x) \log \frac{p(y|x)}{p(y|t)} - \tilde{\lambda}(x) \right] = 0$$

which is equivalent to equation (11) for $p(t|x)$,

$$p(t|x) = \frac{p(t)}{Z(x, \beta)} \exp(-\beta D_{KL}[p(y|x) || p(y|t)])$$

with

$$Z(x, \beta) = \exp[\beta \tilde{\lambda}(x)] = \sum p(t) \exp(-\beta D_{KL}[p(y|x) || p(y|t)])$$

Blahut-Arimoto Algorithm

- The above self consistent equations suggest a natural method for finding the unknown distributions, at every value of β .
- These equations can be turned into converging, alternating iterations

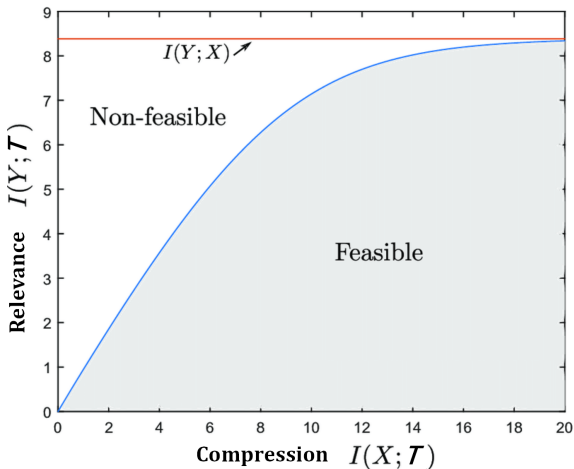
Blahut-Arimoto Algorithm

The minimization is performed by the converging alternating iterations. Denoting by k the iteration step,

$$\begin{cases} p_k(t | x) = \frac{p_k(t)}{Z_k(x, \beta)} \exp(-\beta d(x, t)) \\ p_{k+1}(t) = \sum_x p(x) p_k(t | x) \\ p_{k+1}(y | t) = \sum_y p(y | x) p_k(x | t) \end{cases} \quad (13)$$

Relevance-Compression Region

- The $I(X; T)$ and $I(T; Y)$ can be computed and plotted in the information plane given different β .



Outline

- 1 Background
- 2 Information Bottleneck for Relevance
- 3 Information Bottleneck Views of DNNs
- 4 Information Bottleneck as Optimization Objective**
 - Blahut-Arimoto Algorithm for Known Joint Distribution
 - Variational Information Bottleneck**
- 5 Summary

How to Achieve It for Unknow Distribution?

Challenges:

- Only samples $\{x_i, y_i\}_{i=1}^N$ are available.
- $p(x, y)$ is not known.
- Mutual Information $I(X; T)$ is intractable.

How to Achieve It for Unknow Distribution?

Challenges:

- Only samples $\{x_i, y_i\}_{i=1}^N$ are available.
- $p(x, y)$ is not known.
- Mutual Information $I(X; T)$ is intractable.

Solution:

- Parameterize the encoder function: $T = f_\phi(x)$ in DNNs.
- The minimization of IB Lagrangian reduces to

$$\min_{\phi} I(X; f_\phi(X)) - \beta I(f_\phi(X); Y) \quad (14)$$

- Therefore, minimize $I(X; f_\phi(X))$, maximize $I(f_\phi(X); Y)$

Maximize $I(T; Y)$

Using the fact that the KL-divergence is always positive, we have

$$\text{KL}[p(Y | T), q(Y | T)] \geq 0 \implies \int p(y | t) \log p(y | t) dy \geq \int p(y | t) \log q(y | t) dy$$

and hence,

$$\begin{aligned} I(T, Y) &\geq \int p(y, t) \log \frac{q(y | t)}{p(y)} dy dt \\ &= \int p(y, t) \log q(y | t) dy dt - \int p(y) \log p(y) dy \\ &= - \underbrace{\left(- \int p(y, t) \log q(y | t) dy dt \right)}_{\text{cross-entropy}} + H(Y) \end{aligned}$$

where the entropy of labels $H(Y)$ is independent of the optimization procedure and so can be ignored.

Tractable Upper Bound for $I(X; T)$

When the conditional distribution $p(t|x)$ is known, **upper bounding** $I(X; T)$ is possible [2],

$$\begin{aligned} I(X; T) &\equiv \mathbb{E}_{p(x,t)} \left[\log \frac{p(t|x)}{p(t)} \right] \\ &= \mathbb{E}_{p(x,t)} \left[\log \frac{p(t|x)q(t)}{q(t)p(t)} \right] \\ &= \mathbb{E}_{p(x,t)} \left[\log \frac{p(t|x)}{q(t)} \right] - \text{KL}(p(t) \| q(t)) \\ &\leq \mathbb{E}_{p(x)} [\text{KL}(p(t|x) \| q(t))] \end{aligned} \tag{15}$$

Tractable Upper Bound for $I(X; T)$

- Since we have

$$I(X; T) \leq \mathbb{E}_{p(x)}[\text{KL}(p(t | x) \| q(t))] \quad (16)$$

- This bound is tight when $q(t) = p(t)$.
- A Normal distribution is always utilized for $q(t)$ [2, 5], i.e.,

$$q(t) \sim \mathcal{N}(0, 1)$$

- Minimizing the upper bound can limit the capacity of a stochastic representation.

Tractable Upper Bound for $I(X; T)$

- Since we have

$$I(X; T) \leq \mathbb{E}_{p(x)} [\text{KL}(p(t | x) \| q(t))] \quad (16)$$

- This bound is tight when $q(t) = p(t)$.
- A Normal distribution is always utilized for $q(t)$ [2, 5], i.e.,

$$q(t) \sim \mathcal{N}(0, 1)$$

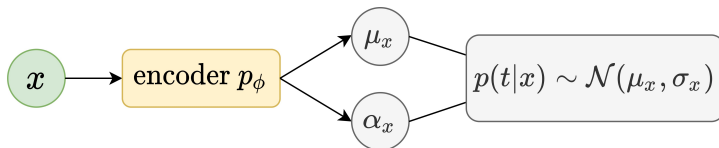
- Minimizing the upper bound can limit the capacity of a stochastic representation.
- Why not other distribution for $q(t)$?

Implementation: Reparameterization Trick

Therefore, minimizing $I(X; T)$ reduces to the following,

$$\min_{\phi} I(X; T) \equiv \min_{\phi} \mathbb{E}_{p(x)} [\underbrace{KL(p_{\phi}(t|x))}_{\text{Encoder}} \parallel \underbrace{q(t)}_{\text{fixed}}]$$

- How to obtain the conditional distribution $p_{\phi}(t|x)$ in DNNs?
- **Reparameterization**: Sample t from the parameterized distribution:



- Finally, solve the following problem,

$$\min \text{KL} [\mathcal{N}(\mu_x, \sigma_x) \parallel \mathcal{N}(0, 1)] \quad (17)$$

Compression Promotes Clustering

The IB curve can be plotted by varying the hyper-parameter β [1, 6].

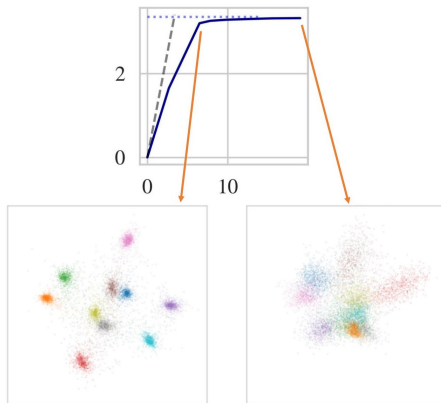


Figure: The IB curve plotted on MNIST dataset and clustering results.

Adversarial Information bottleneck

- **Problem:** The difficulty of optimizing the IB Lagrangian lies in the compression term

$$\min I(X; T) \equiv \min_{\phi} \mathbb{E}_{p(x)} [KL(\underbrace{p_{\phi}(t | x)}_{\text{Encoder}} \| \underbrace{q(t)}_{\text{fixed}})]$$

- **Motivation:** Cancel the underlying assumption of t , $q(t)$ and effectively approximate the KL divergence.
- The KL divergence can be estimated from samples [7].

$$I(X; T) = \max_{T: \Omega \rightarrow R} \mathbb{E}_{p(x,t)} [T(x, z)] - \log \mathbb{E}_{p(x), p(t)} [\exp T(x, z)]$$

Adversarial Information bottleneck

- Adversarial Information bottleneck (AIB) [13] without assumptions about q
- Consider an encoder E_θ and a decoder T_ψ parameterized by θ and ψ , respectively.
- The minimization of the compression term can be , which can be optimized effectively by solving:

$$\min_{\theta} \max_{\psi} \mathbb{E}_{p(x,t)} [T_\psi(x, E_\theta(x))] - \log \mathbb{E}_{p(x), p(t)} [\exp T(x, E_\theta(x))]$$

AIB achieves better IB curves

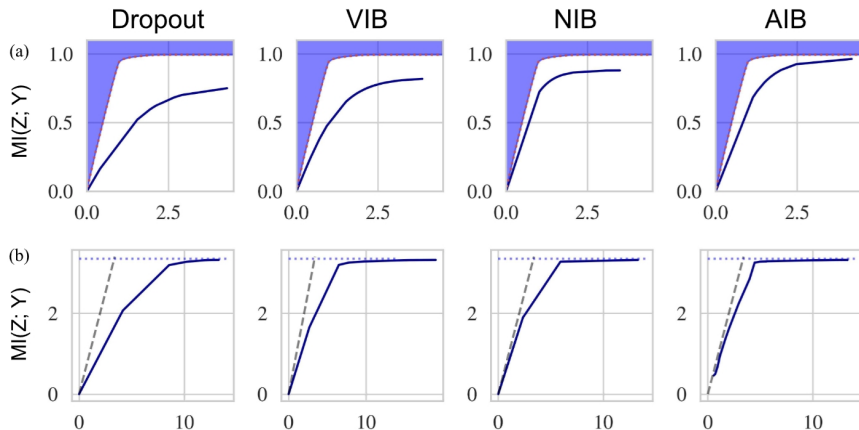


Figure: AIB achieves better IB curves on (a) synthetic data and (b) MNIST dataset

AIB improves the compression optimizaiton

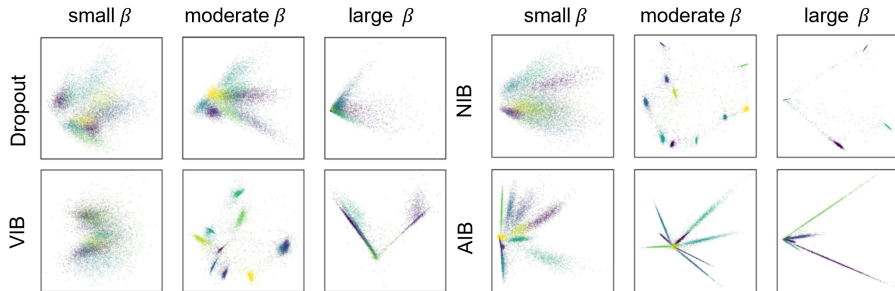


Figure: AIB effectively optimizes the compression terms in MNIST with varying β

Outline

- 1 Background
- 2 Information Bottleneck for Relevance
- 3 Information Bottleneck Views of DNNs
- 4 Information Bottleneck as Optimization Objective
- 5 Summary

Beyond supervised IB learning

- In the unsupervised setting, data samples are not directly labeled by classes
- General forms of **unsupervised IB learning**

$$I_{p(x), p(t|x)}(X; T) - \beta I_{p(t), q(x|t)}(T; \hat{X})$$

- A special case of the supervised IB, where labels are replaced with the reconstruction performance.
- Ziv and LeCun [9] provides a comprehensive review on IB learning.
- Many generative models, including VAE, adversarial autoencoders, InfoVAEs, VAE/GANs can be viewed as unsupervised IB.

Summary

- In DNNs, **relevant information** in X is useful for predicting Y .
- Relevant information are **refined** and redundant information in X are **compressed**.

Summary

- In DNNs, **relevant information** in X is useful for predicting Y .
- Relevant information are **refined** and redundant information in X are **compressed**.
- IB Lagrangian can be taken as **objective** for further compression.
- For known $p(x, y)$, which only exists for very special cases, the IB Lagrangian problem has a closed-form solution.
- For unknown $p(x, y)$, a tractable upper variational bound for $I(X; T)$ can be regarded as a substitute.
- **Appropriate compression** can improve the generalization.

References I



A. Achille and S. Soatto.

Information dropout: Learning optimal representations through noisy computation.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 40:2897–2905, 2018.



A. A. Alemi, I. S. Fischer, J. V. Dillon, and K. Murphy.

Deep variational information bottleneck.
ArXiv, abs/1612.00410, 2017.



R. Bassily, S. Moran, I. Nachum, J. Shafer, and A. Yehudayoff.

Learners that use little information.
In F. Janoos, M. Mohri, and K. Sridharan, editors, *Proceedings of Algorithmic Learning Theory*, volume 83 of *Proceedings of Machine Learning Research*, pages 25–55. PMLR, 07–09 Apr 2018.



T. M. Cover and J. A. Thomas.

Elements of information theory.
1991.



D. P. Kingma and M. Welling.

Auto-encoding variational bayes.
CoRR, abs/1312.6114, 2014.



A. Kolchinsky, B. D. Tracey, and D. H. Wolpert.

Nonlinear information bottleneck.
Entropy, 21:1181, 2019.



S. Nowozin, B. Cseke, and R. Tomioka.

F-gan: Training generative neural samplers using variational divergence minimization.
In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 271279, Red Hook, NY, USA, 2016. Curran Associates Inc.

References II



A. M. Saxe, Y. Bansal¹, J. Dapello, M. Advani, A. Kolchinsky, B. D. Tracey, and D. D. Cox.
On the information bottleneck theory of deep learning.
In *ICLR*, 2018.



R. Shwartz Ziv and Y. LeCun.
To compress or not to compressself-supervised learning and information theory: A review.
Entropy, 26(3):252, 2024.



R. Shwartz-Ziv and N. Tishby.
Opening the black box of deep neural networks via information.
ArXiv, abs/1703.00810, 2017.



N. Tishby, F. C. Pereira, and W. Bialek.
The information bottleneck method.
ArXiv, physics/0004057, 2000.



N. Tishby and N. Zaslavsky.
Deep learning and the information bottleneck principle.
2015 IEEE Information Theory Workshop (ITW), pages 1–5, 2015.



P. Zhai and S. Zhang.
Adversarial information bottleneck.
IEEE Transactions on Neural Networks and Learning Systems, 35(1):221–230, 2022.