INFO8004: A review of the Information Bottleneck in Deep Learning

Eduardo V. Gandara[1] and **Aurélien Werenne**[2]

[1] *e.varas@student.uliege.be (s184314)*
[2] *awerenne@student.uliege.be (s110995)*

ABSTRACT

Deep Neural Networks are used in many applications nowadays, from image classification to speech recognition, achieving human-level performance. Paradoxically, the success of Deep Learning (DL) is far from being fully understood. Why are deep learning models able to generalize? Why is the performance increased when stacking more layers? An attempt to answer those questions is made with the Information Bottleneck theory. The objective of this review paper is i) to present the fascinating results obtained by applying the Information Bottleneck to DL ii) to discuss the limitations of the theory iii) explore potential applications.

I. INTRODUCTION

For the course INFO8004 the assignment of our group was to analyse and summarize the Information Bottleneck (IB) and how it is used to interpret Deep Neural Networks (DNN) [1, 2]. In Section 2 of this paper the necessary background is presented to understand the Information Bottleneck. Section 3 then starts by presenting how the IB is applied to Deep Learning. Next, we explain the main results of the experiments performed by the authors. Lastly, the limitations and possible further applications that could be developed are discussed.

II. BACKGROUND

Mutual Information

Let $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ be two random variables with a joint distribution $p(X, Y)$. The concept of Mutual Information (MI) is defined as

$$I(X;Y) = \mathbb{E}\left[log\left(\frac{p(X,Y)}{p(X)p(Y)}\right)\right] \tag{1}$$

For most distributions (except exponential distributions) direct MI computation is intractable. One common estimation technique is to discretize continuous variables by binning. The expectation expressed in (1) can then be approximated with sampling techniques. It is worth noticing that this binning process introduces some noise.

Minimal Sufficient Statistics

A *sufficient statistic* is defined as the quantity of relevant information on the targets Y contained in the observations X. More specifically, a probabilistic function $S(X)$ is a sufficient statistic for Y if and only if

$$I(S(X);Y) = I(X;Y) \tag{2}$$

Furthermore, a *minimal sufficient statistic* $T(X)$ is the optimal representation containing all the available information about Y, while being the best possible compression of X:

$$T(X) = \underset{S(X)\,:\,I(S(X);Y)=I(X;Y)}{\arg\min} I(S(X);X) \tag{3}$$

Information Bottleneck

Solving the optimization problem (3) is in most practical problems (highly dimensional input space) difficult to compute. Inspired from the Rate-Distortion theory of Shannon, Tishby presented the IB method [3] in order to turn this optimization problem solvable. Using the Lagrange relaxation, Equation (3) is solved like

$$\min_{p(t|x)} I(T;X) - \beta I(T;Y) \tag{4}$$

with β the Lagrange multiplier controlling the trade-off between the amount of information about Y being compressed/preserved .

III. INFORMATION BOTTLENECK AND DEEP LEARNING

Let $X \in \mathcal{X}$ be the inputs and $Y \in \mathcal{Y}$ the ground-truth labels, the authors model the DNN as a Markov Chain where each hidden layer i is represented as a single random variable T_i. Thereupon, they proved experimentally that the hidden layers T_i approach the IB optimal solution for different values of β (see Figure 1).

Information Plane

One of the main contributions is the study of the learning process of neural networks in the information plane, introduced in [1]. Their experiments using the information plane showed how mutual informations $I(T;X)$ and $I(T;Y)$ evolve during the optimization process (see figure and link to video in the original paper). Initially, it can be seen that the deeper layers fail to preserve the information about the input since the layers are randomly initialized. Moreover, Tishby et al. distinguish two phases observed visually. In the first

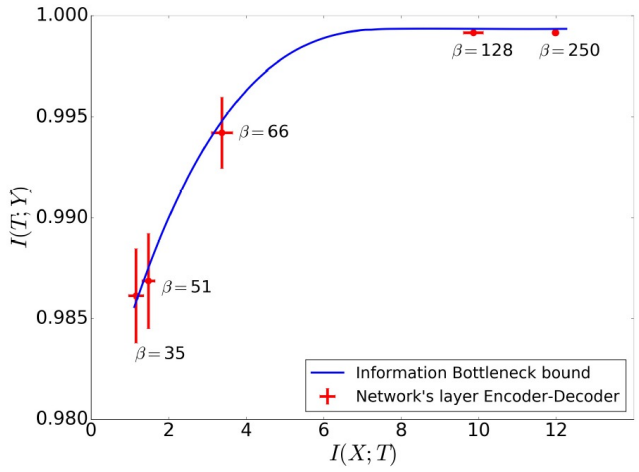

FIG. 1. The DNN layers converge to fixed-points of the IB equations. The error bars represent standard error measures with N=50. In each line there are 5 points for the different layers. For each point, β is the optimal value that was found for the corresponding layer

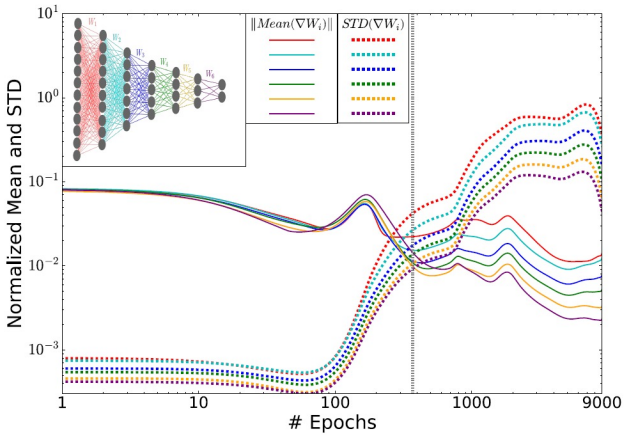

FIG. 2. The layers Stochastic Gradients distributions during the optimization process. The norm of the means and standard deviations of the weights gradients for each layer, as function of the number of training epochs (in log-log scale).

phase of the learning process, the model fits to the data in order to improve its predictions. More interestingly, in the second much slower phase, mutual informations in the information plane move to the upper left corner, in other words, layers lose information about their input while slightly increasing information about the labels. Authors call this phase is the *compression* or *diffusion* phase. They claim that the compression part of the training is responsible for reducing generalization error.

Stochastic Gradient Descent

In our opinion, the most interesting part of the paper is the explanation of how neural networks achieve compression. Figure 2 displays the mean and variance of the gradients per layer. Notice that the two previously mentioned phases can be observed. For the first 300 epochs, the gradients have a high norm and very small deviations, then the variance grows larger until becoming more important than the mean. While the network is fitting the data, the loss is decreasing rapidly. As expected, this corresponds to large gradients with low variance. On the contrary, the compression phase has noisy gradients (i.e. high variance). The authors explain that Stochastic Gradient Descent with a noisy gradient is equivalent to a Random Walk in the parameter space of the loss function. As a consequence, random noise is added to the weights, leading to an increase in the entropy of layers T_i with respect to the inputs. It is worth noticing that this occurs while keeping the training error stable. The conditional entropy $H(X|T_i)$ being increased will cause a decrease of the mutual information $I(X;T_i)$, since the input entropy $H(X)$ remains unchanged.

Overfitting and Hidden Layers

This new insight helps us better understand two well-known phenomena in deep learning: *overfitting* and *deeper models generalize faster*.

Overfitting can be interpreted in the information plane (see Figure 3). We observe that overfitting occurs mostly in the compression phase, where it seems that the representation in the layers is simplified too much.

The computational benefit of deeper NN is according to the authors due to the fact that adding more hidden layers makes the compression go faster, i.e. generalizing faster. Indeed using the Focker-Plank equation [5], the quantity of entropy increase due to SGD follows the relationship

$$\Delta H \propto log(D\tau) \tag{5}$$

with D the diffusion constant and τ the number of epochs. Thus, the number of epoch needed to make a compression ΔI_X grows exponentially as $\exp(\Delta I_X/D)$. If the compression task is shared among K hidden layers, each layer compresses ΔI_X^k, where $\Delta I_X = \sum_k \Delta I_X^k$. Thereupon, the time needed for these K layers to achieve the compression, ΔI_X, is $\sum_k \exp(\Delta I_X^k/D)$. Since the exponential of a sum grows larger than the sum of an exponential, the authors deduce that compression is achieved faster in deeper networks.

IV. LIMITATIONS

A major drawback in this work is that the used model for the experiments only contained saturating activation

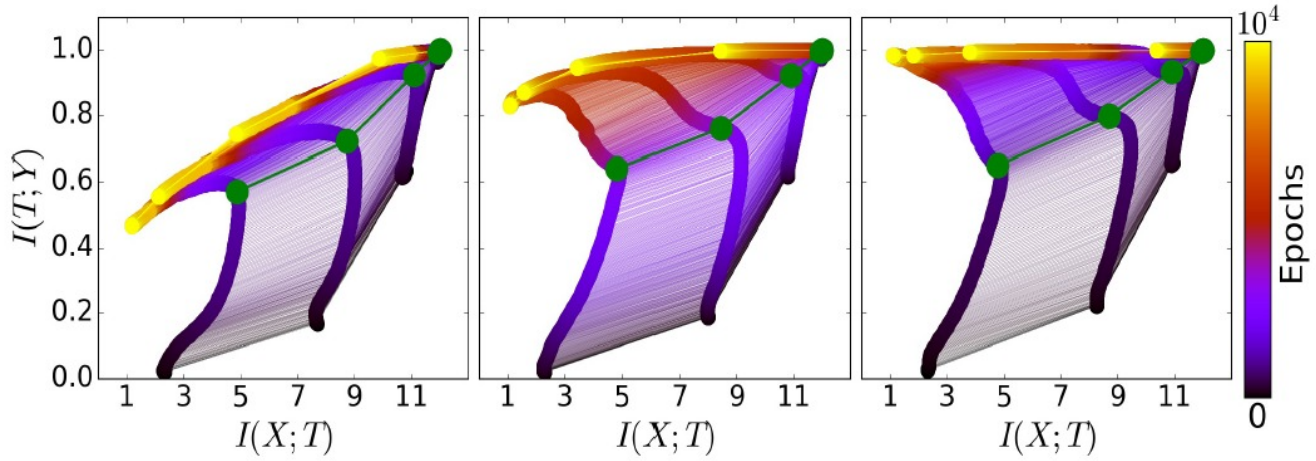

FIG. 3. The evolution of the layers with the training epochs in the information plane, for different training samples. On the left - 5% of the data, middle - 45% of the data, and right - 85% of the data. The colors indicate the number of training epochs with Stochastic Gradient Descent from 0 to 10000. The network architecture was fully connected layers, with widths: input=12-10-8-6-4-2-1=output. The examples were generated by the spherical symmetric rule described in the text.

functions. Saxe et al. [6] showed that neural network using the common ReLU function yield no clear sign of compression. Thereupon, Tishby replied that their methodology for mutual information estimation is incorrect since non-saturating functions may lead to unbounded hidden activity. And thus, the level of noise brought by the binning process (see Section II) becomes inconsistent. To counter this undesired effect, Chelombiev et al. [7] developed an adaptive estimator, which showed that the fast fitting phase is indeed followed by a slow compression phase occurs for ReLU function.

An important issue preventing the IB to be easily integrated in a DL setting is that mutual information estimation is computationally expensive in real-world machine learning problems. Possible solutions involve developing faster estimators [7, 9] or as suggested by Tishby in one of his talks, using the order of magnitude of the proportion between mean and variance of the gradients as a proxy for MI estimation.

In [10] a very interesting relationship between Variational Auto-encoders (VAE) [11] and the Information Bottleneck was discovered. If we replace $I(T;Y)$ by $H(Y) - H(Y|T)$, the IB optimization (4) becomes

$$\min_{p(t|x)} I(X;T) + \beta H(Y|T) \tag{6}$$

Moreover,

$$H(Y|T) = \mathbb{E}_{p(x,y)}\left[\mathbb{E}_{p(t,X)}\left[-log(p(Y|T))\right]\right] \tag{7}$$

$$I(X|T) = \mathbb{E}_{p(x)}\left[\mathrm{KL}(p(t|X)||p(t)\right] \tag{8}$$

If those expressions are substituted in Eq.6, the IB can be approximated by a loss function as,

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{p(t|x^{(i)})}\left[-log(p(y^{(i)}|t))\right] + \beta\,\mathrm{KL}(p(t|x^{(i)})||p(t))$$

This equation is equivalent to the VAE loss in the special case when $\beta = 1$. It was shown that using a VAE with a hyperparameter $\beta > 1$ beats several state-of-the art results. This relationship, in our opinion conforts the justification of using MI as a tool to analyze or improve DL techniques.

It is also worth noting that noise plays an important role in compression, and thus improving generalization. One could imagine using noise to improve existing optimization algorithms or even develop new ones.

[1] Naftali Tishby and Noga Zaslavsky. Deep Learning and the Information Bottleneck Principle. *arXiv e-prints*, page arXiv:1503.02406, Mar 2015.

[2] Ravid Shwartz-Ziv and Naftali Tishby. Opening the Black Box of Deep Neural Networks via Information. *arXiv e-prints*, page arXiv:1703.00810, Mar 2017.

[3] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. *arXiv e-prints*, page physics/0004057, Apr 2000.

[4] Data Processing Inequality. Data processing inequality — Wikipedia, the free encyclopedia, 2010. [Online; accessed 29-September-2012].

[5] Fokker-Planck equation. Fokker-planck equation — Wikipedia, the free encyclopedia, 2010. [Online; accessed 29-September-2012].

[6] Andrew et al. M. Saxe. On the information bottleneck theory of deep learning. *ICLR 2018*, 2015.

[7] Ivan Chelombiev, Conor Houghton, and Cian O'Donnell. Adaptive Estimators Show Information Compression in Deep Neural Networks. *arXiv e-prints*, page arXiv:1902.09037, Feb 2019.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[9] Morteza Noshad, Yu Zeng, and III Hero, Alfred O. Scalable Mutual Information Estimation using Dependence Graphs. *arXiv e-prints*, page arXiv:1801.09125, Jan 2018.

[10] Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep Variational Information Bottleneck. *arXiv e-prints*, page arXiv:1612.00410, Dec 2016.

[11] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv e-prints*, page arXiv:1312.6114, Dec 2013.