Cross-Entropy Regularization with Mutual Information in Training CNNs

Zaugg, Christoph; Ingold, Rolf; Trendafilov, Dari Borisov; Fischer, Andreas

DOI:
10.1007/978-3-031-93631-9_16

Publication date:
2025

Document Version
Early version, also known as pre-print

Cross-Entropy Regularization with Mutual Information in Training CNNs

Zaugg, Ch.[1,3], Ingold, R.[3], Trendafilov, D.[2], and Fischer, A.[3,4]

[1] ZHAW, ACSS, Technikumstrasse 9, 8400 Winterthur, Switzerland,christoph.zaugg@zhaw.ch
[2] Namur Institute for Complex Systems, University of Namur, Namur, Belgium, dari-borisov.trendafilov@unamur.be
[3] Department of Informatics, DIVA, Boulevard de Pérolles 90, 1700 Fribourg University, Switzerland,{christoph.zaugg, rolf.ingold, andreas.fischer}@unifr.ch,
[4] iCoSys, HES-SO, Boulevard de Pérolles 80, 1700 Fribourg, Switzerland

Abstract. We examine the learning behavior of a shallow and a deep convolutional neural network performing classification tasks on subsets of two databases. Our investigation focuses on the label, the input, and the prediction layer, and we compute the mutual information between these layers epoch-wise using Rényi's matrix-based entropy functional. We evaluate the data processing inequality to interpret the learning behavior in a consistent information-theoretic framework. Our primary goals are to 1) clarify the relation between the two training objectives of minimizing the cross-entropy and maximizing the mutual information between the label and the prediction layer, 2) gradually switch from the first to the second training objective, and 3) interpret the impact of the latter transition. One of the main contributions is the proposed novel method for regularizing the cross-entropy objective and assessing the neural network's learning activity.

Keywords: Neural networks · Rényi's entropy functional · Data processing inequality · InfoMax

1 Introduction

Recent advances in deep learning provide evidence that with enormous datasets and computing power, arbitrary big models could offer unprecedented performance. However, this learning process has the well-known downside of over-fitting. More recent advances aim to eliminate this bias and optimize learning using sparse datasets and a shorter training process, also known as one-shot learning [16, 10].

This paper presents an initial study exploiting generic information-theoretic functionals to optimize convolutional neural networks (CNNs) and identify model saturation over time. The aim is to speed up the training process and mitigate over-fitting.

In training two types of CNNs in classification tasks on two databases, we focus on the label layer Y, the input layer X, and the prediction layer $\hat{Y}$ and epoch-wise compute the mutual information (MI) values $I(Y;X)$, and $I(Y;\hat{Y})$. We compare both values to assess the data processing inequality (DPI):

$$I(Y;X) \geq I(Y;\hat{Y}). \tag{1}$$

The DPI is a necessary condition in an information-theoretic framework and serves to cross-check the approximated MI values. In [19], the authors investigate chains of DPIs to optimize the learning behavior of a Multilayer Perceptron.

We clarify the relation between two training objectives, i.e., minimizing cross-entropy (CE) and maximizing $I(Y;\hat{Y})$. Our loss function – a mixture of CE and $I(Y;\hat{Y})$ – depends dynamically on the training epoch, which brings the benefit of simultaneously minimizing CE and maximizing $I(Y;\hat{Y})$. The resulting insight could provide an automated trigger for terminating the training and avoid potential over-fitting.

We base our reasoning on two pillars: an approximation method for computing MI values between high-dimensional continuous random variables and a specific parameter optimization procedure.

In Section 2, we embed our work into related work, and in Section 3 describe the details of the methodology. In Section 4, we present the setting of our experiments, and Section 5 contains the results about the two training objectives. In Section 6, we discuss our results, and state our conclusions.

2 Related Work

The information bottleneck (IB) principle in [13], or the dimpled manifold model (DMM) in [11] are paradigms to explain the learning behavior of deep neural networks (DNNs). The first stems from information theory and attributes learning to the optimal trade-off between compression and predictability. The second takes a geometrical perspective. The dimension of an image manifold in a classification task tends to be much lower than the dimension of the embedding space. After a few training epochs, the decision boundary is close to the image manifold, and additional training only causes tiny dimples on the decision boundary to bring incorrectly classified data points to the correct side.
Our study is founded on the framework of information theory. It elucidates the learning behavior of a CNN performing a classification task in an early training phase after transient behavior caused by random initialization has subsided.
The authors of [6] estimate and maximize $I(X;\hat{Y})$ mainly for unsupervised learning tasks, i.e., their training objective is Linsker's InfoMax principle stated in [7]. They base their Deep InfoMax (DIM) on the Mutual Information Neural Estimator (MINE) introduced in [1] that provides lower bounds for estimating MI values, and the work in [8] tightens these bounds. In [14], the authors maximize $I(Y;\hat{Y})$ by estimating a probability density function (PDF) using a kernel density and propose $I(Y;\hat{Y})$ as a regularization term for hinge, squared, logistic and exponential loss functions. In our study, we investigate how maximizing

$I(Y;\hat{Y})$ relates to minimizing cross-entropy.
In [4], the authors present Rényi's matrix-based entropy functional of order α and its convergence to the MI as the order α tends to one. The method only relies on estimating the kernel width of the radial basis function (RBF) and does not require a PDF. The authors of [18] estimate the kernel width based on Siverman's rule of thumb [12] and use Rényi's matrix-based entropy functional to approximate MI values between various layers of DNNs up to $50'000$ training epochs. In [15], the authors optimize the kernel width by maximizing the alignment between matrices as proposed in [2]. They train the DNNs up to $5'000$ training epochs and elaborate on the compliance of the DPIs.
Our study, investigating the early training phase of two types of CNNs, aims at identifying a learning saturation criterion that could prevent over-fitting. The significant fluctuation in the MI values poses a major challenge during that phase. Instead of following MINE's lower bounds, we resorted to Rényi's matrix-based entropy functional, which is known to converge. The consistency of the DPIs serves as a validation guideline.

3 Methodology

3.1 Approximation

We apply Rényi's matrix-based entropy functional of order α [18] to the layers Y, X, and $\hat{Y}$. Considering a batch a of size N in one of these layers, we compute the kernel matrix from the row vectors a_i as follows:

$$K_{ij} = \exp\left(-\frac{\|a_i - a_j\|^2}{s^2}\right). \tag{2}$$

Normalizing the kernel matrix K ensures the data processing inequality (Eq. 1):

$$A_{ij} = \frac{K_{ij}}{\sqrt{A_{ii}} \cdot \sqrt{A_{jj}}}. \tag{3}$$

We compute the contribution of the batch to the entropy $H_\alpha(A)$ using $\alpha = 1.01$:

$$H_\alpha(A) = \frac{1}{1-\alpha} log_2\left(tr(A^\alpha)\right). \tag{4}$$

In Eq. 5 and 6, the matrix B corresponds to another layer and is obtained by applying Eq. 2 and 3 to the batch b. We approximate the joint entropy $H_\alpha(A,B)$ by inserting the trace normalized Hadamard product of A and B:

$$H_\alpha(A,B) = H_\alpha\left(\frac{A \circ B}{tr(A \circ B)}\right) \tag{5}$$

and compute $I_\alpha(A;B)$ as follows:

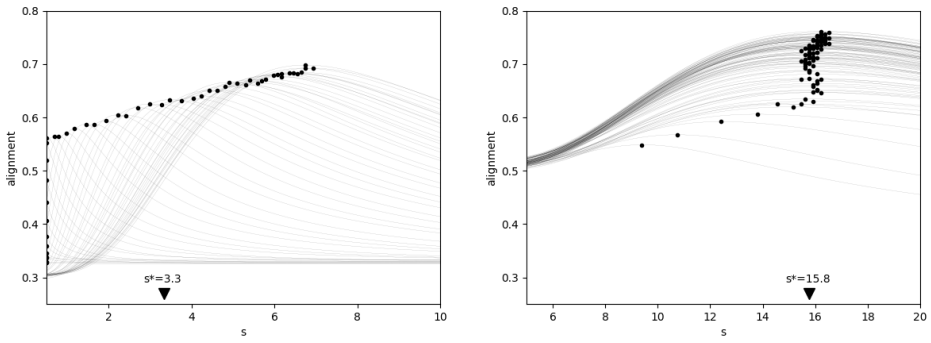
$$I_\alpha(A;B) = H_\alpha(A) + H_\alpha(B) - H_\alpha(A,B). \tag{6}$$

Averaging over the batches approximates the MI between the layers considered, e.g., Y and X, or Y and $\hat{Y}$.

3.2 Optimal Kernel Width

Computing the kernel matrix in Eq. 2 requires estimating the kernel width s of the RBF. The $(i,j)^{th}$ entry of the label matrix takes the value one if the i^{th} and the j^{th} labels in a batch are equal, and zero elsewhere. The label matrix captures the clustering of the labels in a batch.
To make the clustering in another layer resemble as much as possible, Cristianini [2] proposes the kernel width s^* that maximizes the cosine, i.e., the alignment between the label and the kernel matrix associated with the other layer.

In all of our experiments, we maximize the cosine between the label layer Y and kernel matrix of the prediction layer $\hat{Y}$. While a CNN learns, the alignment increases during training, which makes the optimal kernel width s^* depend on the training epoch. On the other hand, the value of s^* sets the scale in Eq. 2 for computing the MI values. With our tools, we can interpret the learning behavior in a consistent information-theoretic framework only during the epochs in which s^* stays within a typical value over a particular observation horizon, e.g., 100 training epochs in Fig. 1(b). Our experiments reveal that the DPI (cf. Eq. 1) is valid during the observation horizon at the beginning of the training. Fig. 1 illustrates the choice of the optimal kernel width s^* in our experiments. Section 4 describes in more detail both types of CNNs (shallow and ResNet18) and the types of databases (Mnist and Fashion-mnist) we experimented with.



(a) Shallow CNN, Mnist, 50 epochs (b) ResNet18, Fashion-mnist, 100 epochs

Fig. 1: Choice of optimal kernel width: We partition the displayed intervals on the abscissa into 100 equidistant kernel widths candidates. Each training epoch is represented with a grey curve whose ordinate value is the batch mean of the alignments, i.e., the cosine values between the label matrix and the kernel matrix of the prediction layer evaluated at the candidate kernel width. The black dots locate the maxima on the grey lines. The optimal parameter s^* is the average of the black dots' abscissa values.

4 Experiments

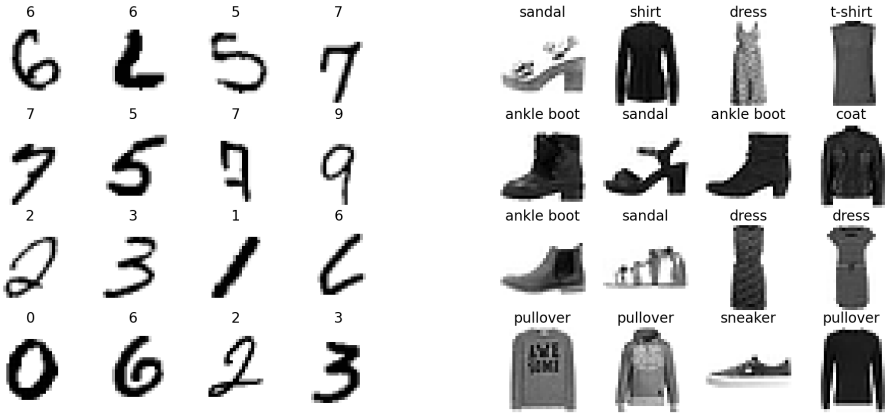
4.1 Data Sets

We experimented with labeled images of the Mnist [3] and the Fashion-mnist [17] databases. The examples consist of ten categories, with the image resolution (1, 28, 28). In order to keep the computational complexity low in this initial study, we extracted rather small balanced subsets of 5'000 training and 1'000 validation and test examples from these databases. Fig. 2 presents random samples of 16 labeled images.

4.2 CNN architectures

We study two CNNs architectures, a shallow one and ResNet18 [9]. The shallow CNN contains four hidden layers. The first two layers perform a convolution followed by a max pooling, and the last two are fully connected linear layers. A rectified linear (ReLU) activation function terminates each hidden layer. The prediction layer, that assigns the label with the highest score to an image, is fully connected. The shallow CNN has approximately 44K network parameters.

To contrast the shallow CNN, we also experimented with one type of ResNet. The authors in [5] partition the layer sequence of a CNN into blocks. Instead of training the network parameters of the entire layer sequence, they train the parameters of the residual functions assigned to each block. We studied ResNet18 [9] with roughly 200 times as many network parameters as the shallow CNN. In the experiments with ResNet, we use images with resolution (3, 224, 224).



(a) Mnist (b) Fashion-mnist

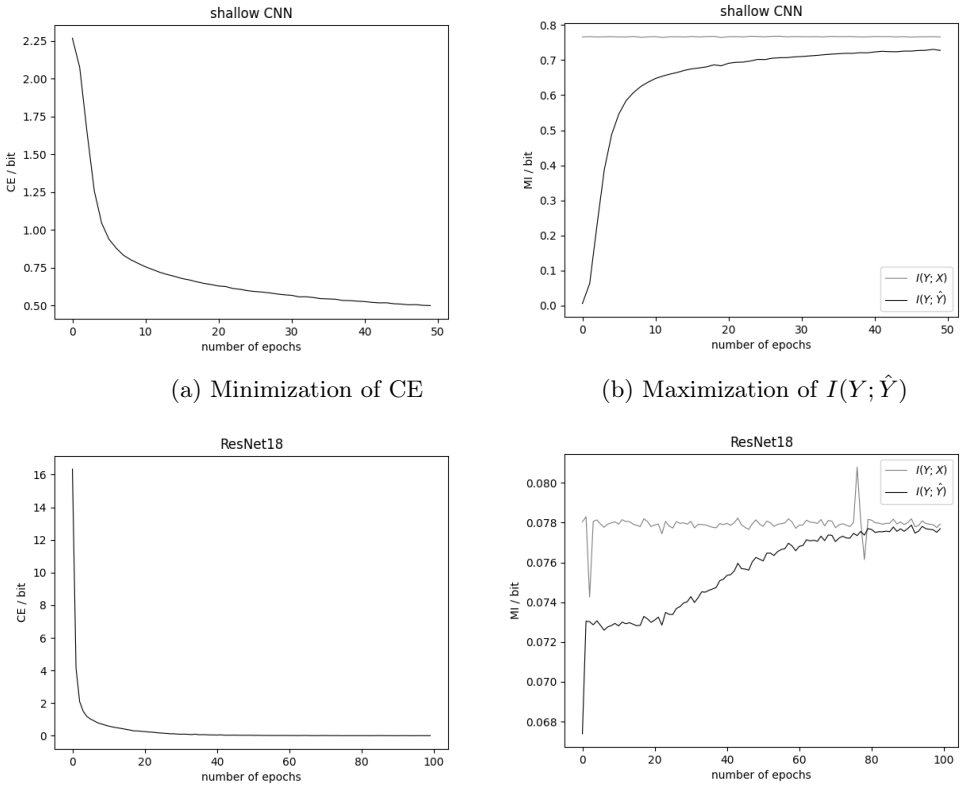
Fig. 2: Random samples of the two databases Mnist and Fashion-mnist, used in our experiment, with resolution of (1, 28, 28).

5 Results

5.1 Approximate Equivalence

Our results (see Fig. 3) on the Fashion-mnist data reveal that minimizing cross-entropy (CE) maximizes the mutual information ($I(Y;\hat{Y})$) between the label Y and the prediction $\hat{Y}$ layers. This trend holds for the shallow CNN trained over 50 epochs (cf. Fig. 3ab) and for ResNet18 trained over 100 epochs (cf. Fig. 3cd). The grey curves in Fig. 3bd denote $I(Y;X)$. We observed that the data processing inequality (Eq. 1) is violated once (cf. Fig. 3d), possibly due to a numeric instability. The experimental results for the Mnist data are similar.

The maximization of $I(Y;\hat{Y})$ raises the problem of a vanishing gradient for both CNNs. To avoid that we use CE as an ignitor, i.e., we introduce a transition



(a) Minimization of CE

(b) Maximization of $I(Y;\hat{Y})$

(c) Minimization of CE

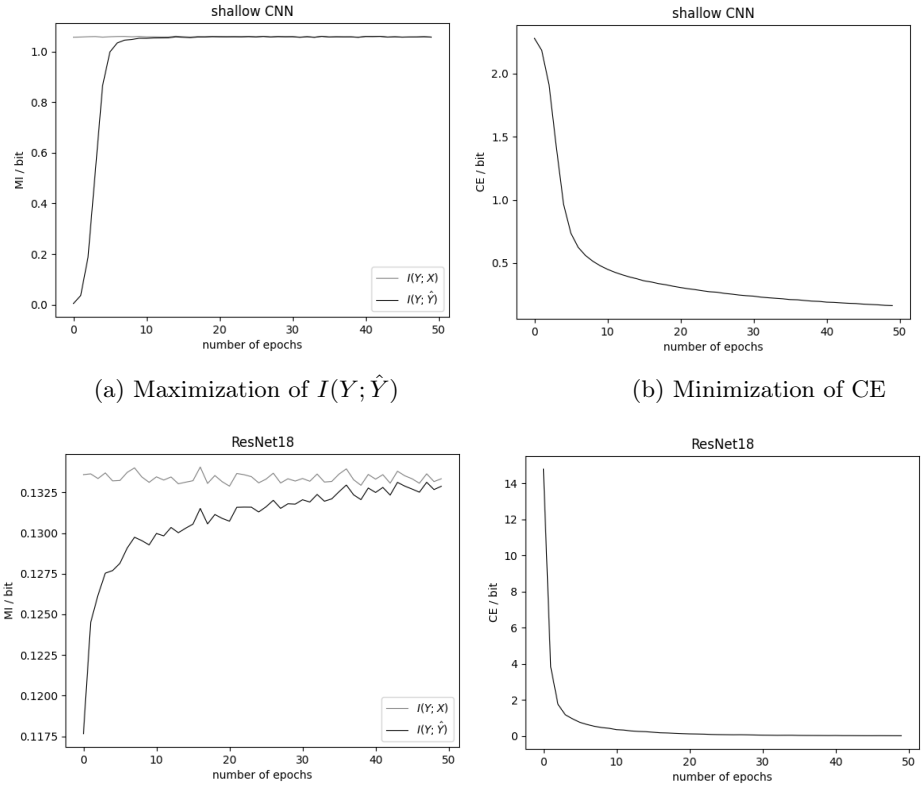
(d) Maximization of $I(Y;\hat{Y})$

Fig. 3: Minimizing CE implies maximizing $I(Y;\hat{Y})$ on the Fashion-mnist data. In (a), we minimize the CE of the shallow CNN and observe in (b) that $I(Y;\hat{Y})$ increases. In (c), we minimize the CE of ResNet18 and observe in (d) that $I(Y;\hat{Y})$ increases. The results on the Mnist data for both types of networks are similar.

factor f that depends on the training *epoch* such that the training objective L at $epoch = 0$ coincides with CE and gradually allows $I(Y;\hat{Y})$ to gain influence as the number of training epochs grows:

$$L = f(epoch) \cdot CE - (1 - f(epoch)) \cdot I(Y;\hat{Y}). \tag{7}$$

Fig. 4 presents the results from minimizing the loss L (Eq. 7) with $f(epoch) = (1+epoch)^{-0.5}$ on the Mnist training data. We observed that any transition factor f depending on the training epoch that lets L coincide with the cross-entropy objective at the beginning of the training and then gently allows $I(Y;\hat{Y})$ to gain influence yields a non-vanishing gradient of $I(Y;\hat{Y})$, even though the gradient at $epoch = 0$ vanishes as Fig. 6ab reveal. Hence, we can approximately maximize



(a) Maximization of $I(Y;\hat{Y})$

(b) Minimization of CE

(c) Maximization of $I(Y;\hat{Y})$

(d) Minimization of CE

Fig. 4: Maximizing $I(Y;\hat{Y})$ minimizes CE via Eq. 7 on the Mnist data: In (a), we maximize $I(Y;\hat{Y})$ of the shallow CNN and observe in (b) that the CE decreases. In (c), we maximize $I(Y;\hat{Y})$ of ResNet18 and observe in (d) that the CE decreases. The results on the Fashion-mnist data for both types of networks are similar.

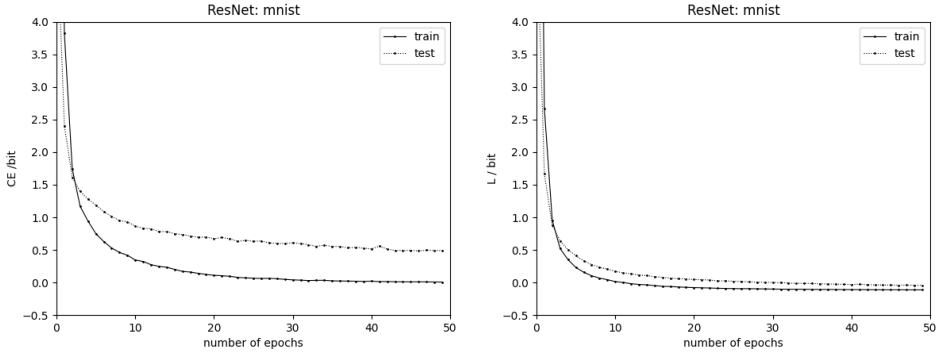
$I(Y;\hat{Y})$ and minimize CE in parallel. Fig. 4ab reflects this for the shallow CNN and Fig. 4cd for ResNet18, respectively.

5.2 Cross-Entropy Regularization

We interpret Eq. 7 as a novel method to regularize cross-entropy with the mutual information between the label and the prediction layer. In Fig. 5, we illustrate the effect of training ResNet18 on Mnist. If we minimize CE, the test loss remains above the training loss (see Fig. 5 (a)). If we minimize L the test and the training loss almost coincide (see Fig. 5 (b)).

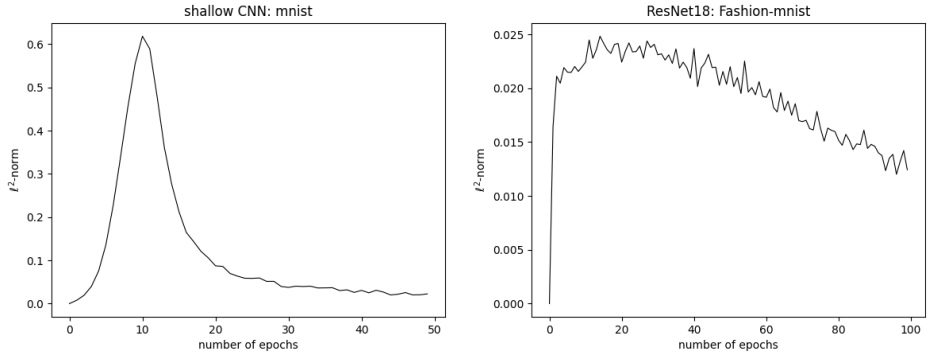
5.3 Saturation of $I(Y;\hat{Y})$

Our experiments demonstrate that $I(Y;\hat{Y})$ increases with ongoing training. In an information-theoretic interpretation, the network reduces the uncertainty in the prediction layer as much as possible, while observing the label layer. The saturation of $I(Y;\hat{Y})$ is a natural criterion to asses the learning activity. We can approximately compute this quantity and its gradient. Fig. 6 presents the ℓ^2 norm of the gradient $I(Y;\hat{Y})$ to characterize the learning of shallow CNN on the Mnist data. In Fig. 4 (a), we see that $I(Y;\hat{Y})$ saturates after just a few training epochs. The bump in Fig. 6 (a) corroborates the observation. ResNet18 on Fashion-mnist shows a more extended learning activity in Fig. 4 (c), and Fig. 6 (b) shows its potential for additional learning.



(a) Minimization of CE (b) Minimization of L

Fig. 5: Regularization property on the Mnist data. In (a), we minimize CE and observe a persistent gap between training and test loss. In (b), we minimize L according to Eq. 7 and observe how the gap closes. The results on the Fashion-mnist data for both types of networks are similar.



(a) $I(Y;\hat{Y})$ (b) $I(Y;\hat{Y})$

Fig. 6: The gradient of $I(Y;\hat{Y})$ in training CNNs with L according to Eq. 7. In (a), the gradient of $I(Y;\hat{Y})$ quickly vanishes for the shallow CNN trained on the Mnist data. The result is consistent with a fast saturation of $I(Y;\hat{Y})$ in Fig. 4 (a), revealing that learning has come to an end. In (b), the gradient of $I(Y;\hat{Y})$ slowly decreases for ResNet18 trained on the Fashion-mnist data and keeps updating $I(Y;\hat{Y})$, revealing that learning goes on.

6 Discussion and Conclusion

Using Rényi's matrix-based entropy functional to approximate $I(Y;\hat{Y})$ causes its gradient to vanish. We overcome this difficulty by leveraging cross-entropy as an ignitor, as we propose in Eq. 7. Our results confirm the interpretation of the second term in Eq. 7 as regularization of cross-entropy. We have a similar result for a different loss function as the authors of [14]. The key difference is that our regularization depends dynamically on the training epoch. If we minimize the loss according to formula Eq. 7, we simultaneously reduce the cross-entropy and increase $I(Y;\hat{Y})$. Maximizing $I(Y;\hat{Y})$ only minimizes the difference between the label and prediction distributions and ignores the quality of the representations in the layers.

With our method, we can observe CNNs' learning behavior at the beginning of the training phase in a consistent information theoretical framework. Using the mutual information between the label and the prediction layer is a novel cross-entropy regularization method that could mitigate over-fitting. Future work will investigate the proposed regularization's impact on the generalization property with larger data sets.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101034383.

References

1. Belghazi, M.I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., Hjelm, R.D.: Mine: mutual information neural estimation. arXiv preprint arXiv:1801.04062 (2018)
2. Cristianini, N., Shawe-Taylor, J., Elisseeff, A., Kandola, J.: On kernel-target alignment. Advances in neural information processing systems **14** (2001)
3. Deng, L.: The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine **29**(6), 141–142 (2012)
4. Giraldo, L.G.S., Rao, M., Principe, J.C.: Measures of entropy from data using infinitely divisible kernels. IEEE Transactions on Information Theory **61**(1), 535–548 (2014)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
6. Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., Bengio, Y.: Learning deep representations by mutual information estimation and maximization. arXiv preprint arXiv:1808.06670 (2018)
7. Linsker, R.: Deriving receptive fields using an optimal encoding criterion. Advances in neural information processing systems **5** (1992)
8. Mroueh, Y., Melnyk, I., Dognin, P., Ross, J., Sercu, T.: Improved mutual information estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 9009–9017 (2021)
9. PyTorch: Resnet18 model. https://pytorch.org/vision/2.0/models/generated/torchvision.models.resnet18.html (2024), accessed: 2024-08-31
10. Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., Lillicrap, T.: Meta-learning with memory-augmented neural networks. In: International conference on machine learning. pp. 1842–1850. PMLR (2016)
11. Shamir, A., Melamed, O., BenShmuel, O.: The dimpled manifold model of adversarial examples in machine learning. arXiv preprint arXiv:2106.10151 (2021)
12. Silverman, B.W.: Density estimation for statistics and data analysis. Routledge (2018)
13. Tishby, N., Zaslavsky, N.: Deep learning and the information bottleneck principle. In: 2015 ieee information theory workshop (itw). pp. 1–5. IEEE (2015)
14. Wang, J.J.Y., Wang, Y., Zhao, S., Gao, X.: Maximum mutual information regularized classification. Engineering Applications of Artificial Intelligence **37**, 1–8 (2015)
15. Wickstrøm, K., Løkse, S., Kampffmeyer, M., Yu, S., Principe, J., Jenssen, R.: Information plane analysis of deep neural networks via matrix-based renyi's entropy and tensor kernels. arXiv preprint arXiv:1909.11396 (2019)
16. Woodward, M., Finn, C.: Active one-shot learning. arXiv preprint arXiv:1702.06559 (2017)
17. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
18. Yu, S., Principe, J.C.: Understanding autoencoders with information theoretic concepts. Neural Networks **117**, 104–123 (2019)
19. Zaugg, C., Ingold, R., Füchslin, R., Fischer, A.: How to turn a leaky learner into a sealed one. In: Artificial Life and Evolutionary Computation: 17th Italian Workshop, WIVACE 2023, Venice, Italy, September 6-8, 2023, Revised Selected Papers. p. 29. Springer Nature (2024)