An Overview of Convergence Analysis of Deep Neural Networks and the Visual Explanations of their Predictions

Mohammed Nowaz Rabbani Chowdhury, RIN: 662011081, email: chowdm2@rpi.edu

***Abstract*—In this report, we reviewed the theoretical and practical aspects of explaining the success of deep neural networks (DNN). The review is centered around two papers, one representing the issue of provable global convergence of the learning algorithms used in DNN [1], other representing a practical approach to explain the predictions generated by them [2]. In [1], a modified learning algorithm has been proposed which can provide provable global convergence guarantee of DNN beyond the so-called NTK regime. In [2], a generalized version of class activation map (CAM) has been introduced which can provide improved class-discriminative property, object localization ability, reliability and interpretability. Some potential future research directions in both domains have been provided in this review.**

I. Introduction

From the theoretical perspective, explaining the success of deep neural network (DNN) is governed by the goal of providing theoretical guarantee for convergence of gradient descent (GD) or stochastic gradient descent (SGD) and providing the guarantee of generalization ability of the learned networks. One of the major progresses in this direction is to answer following two broad questions [3]:

1) Why does the overparametrized DNN converge to global minima despite the optimization problem is highly nonconvex?
2) Why does the overparametrized DNN generalize despite the potential possibility of overfitting?

There are several recent works contributed significantly to answer those questions [3], [4], [5], [6], [7]. Most of these works depend on the assumption that the network is highly overparametrized (i.e., number hidden nodes in each layer is in the order of a large polynomial of number of samples). This leads to the so-called theory of Neural Tangent Kernel (NTK).

The idea behind the NTK theory is that, when the model is highly overparametrized (i.e., in the NTK regime) and the learning rate is sufficiently small, the activation pattern of the hidden nodes remains approximately same throughout the training process compared to the initialization [8], [9]. Then, the first order approximation of the network is valid, and the network becomes approximately linear with respect to weights. This leads to the optimization problem to be convex, and the SGD/GD enjoys linear convergence rate to converge to the global minima.

However, one of the major drawbacks of NTK theory is that it can't explain the capability of DNN to learn representations of multiple abstraction levels from the data as the model in the NTK regime is approximately linear [1]. Also, the degree of overparameterization required to achieve the global convergence makes it incomparable to the practical overparameterized DNN.

To address these issues, in [1] a modified version of the basic (e.g., SGD/GD) optimization algorithm has been proposed which can provably achieve the global minima of the objective function under an assumption stated as *expressivity condition*. The theoretical verification of the condition has been provided for fully connected DNN and a numerical verification of the condition has been provided for Resnet with batch normalization for different benchmark datasets.

The major advantage of the modified algorithm is that it can demonstrates to learn representations from the data and provably achieve the global minima simultaneously without sacrificing the generalization performance compared to the baseline algorithms. Also, the degree of overparameterization required to achieve the convergence is in the linear order of number of samples which makes it comparable to overparametrized DNNs used in practice. However, the lack of provable generalization guarantee can be identified as a major shortcoming compared to NTK approximation of DNN.

One of the major practical aspects of explaining DNN is centered around interpreting its success in vision tasks. However, most of the networks deployed in vision tasks contain several convolution layers. Hence, interpreting such networks has been driven by visualizing the concepts learned by those convolution layers.

Visual explanation from Convolutional Neural Networks (CNN) can be broadly divided into two major

categories [2]: *Pixel-space Gradient Visualization* and *Localization based Visual Explanation*. All of the pixel-space gradient visualization techniques found in literature (e.g., *Backpropagation* [10], *Deconvolution* [11] and *Guided-backpropagation* [12]) provide visualization with a certain degree of high-resolution but lack class-discriminative ability. On the other hand, one of the earliest localization based visual explanation technique found in literature is the *Class Activation Map* (CAM) [13], which can demonstrates high class-discriminativeness but without retraining the technique is only applicable to a very specific network architecture (i.e., global average pooling of the last convolution layer followed by the output layer).

To make the localization based technique more general and applicable to any network without alteration of the structure, a gradient based class activation map has been proposed in [2] known as *Grad-CAM*. Moreover, to leverage the high-resolution capability of pixel-space gradient visualization techniques, guided backpropagation is integrated with Grad-CAM to generate *Guided Grad-CAM* which can demonstrates class-discriminativeness and generates high-resolution map simultaneously. The Grad-CAM and Guided Grad-CAM not only provide visual explanation for prediction of the networks used in classification tasks, but also other vision tasks such as image captioning and visual question answering (VQA). Moreover, it can demonstrate better localization ability compared to other weakly-supervised localization techniques, better human reliability compared to guided backpropagation, can identify failure modes of a network and identify dataset biases.

The rest of the report is structured as follows. Section II provides the contemporary theories of global convergence guarantee of DNN. Section III provides a comparative discussion on different visual explanation techniques from DNN. Section IV provides some potential future research directions in both of these domains. Finally, Section V completes the report with some concluding remarks.

II. Global Convergence Guarantee of Deep Neural Networks

A. Problem Formulation

The optimization problem of a DNN can be formulated as follow:

$$\min_{w} \mathcal{L}(w) = \frac{1}{n}\sum_{i=1}^{n} l(f(x_i, w), y_i) \tag{1}$$

where $\{x_i, y_i\}_{i=1}^{n}$ are the training samples, w contains the learnable parameters of the network, $l(f(x_i, w), y_i)$ is the loss function and $f(x_i, w)$ is the network output function for the i-th training sample. However, due to the non-linearity introduced in the hidden layers by the incorporation of the non-linear activation function turns the minimization problem into a non-convex problem [8]. As a result, typical convex optimization theory is not applicable to prove the global convergence of the algorithms used in practice (e.g., GD, SGD).

B. NTK Based Global Convergence

Although, the objective function represented in (1) is highly non-convex in parameter space (i.e., non-convex w.r.t. w), as most of the loss functions are convex in functional space, the objective function is also convex in functional space (i.e., convex w.r.t. $f(x_i, w)$). Hence, primarily the NTK based convergence analysis depends on the dynamics of the objective function in functional space [8], [9]. Moreover, the analysis depends on the fact that, the parameter (w) of a highly overparameterized (number of hidden nodes is in the polynomial order of number of samples) and properly initialized network remains close to its initial value during the training using GD/SGD for a sufficiently small learning rate. Based on the above observations, it can be shown that, the training of the network can be represented by a kernel method which is convex in parameter space. The kernel is defined as:

$$\mathrm{K}_w(x_i, x_j) = \left(\frac{\partial f(x_i, w)}{\partial w}\right)^T \frac{\partial f(x_i, w)}{\partial w} \tag{2}$$

which is known as Neural Tangent Kernel (NTK) and the corresponding kernel matrix known as NTK matrix can be defined as:

$$\mathbf{K}(w) = \frac{\partial vec(f_X(w)^T)}{\partial w}\left(\frac{\partial vec(f_X(w)^T)}{\partial w}\right)^T \tag{3}$$

Here, $f_X(w) \in \mathbb{R}^{(n \times m_y)}$ is the output matrix of the network, i-th row of which is representing the output of the network for the sample x_i and m_y is the number of nodes at the output layer. Accordingly, the convergence of GD/SGD can be proved with linear convergence rate as long as the NTK matrix is positive-definite (i.e., full-rank) throughout the training process. However, based on a mild assumption on the input data distribution, it can be shown that the NTK matrix is full-rank at initialization. Then, due to the proximity of the parameters during training process to the initialization, the NTK matrix remains full-rank throughout the training process.

C. Global Convergence Guarantee Beyond NTK Regime

To prove the global convergence beyond the NTK regime, in [1], a modified algorithm has been proposed which contains two phases. In the first phase of the algorithm, the network learns the representations of the data while in the second phase, the learning rate has been modified so that it enters in the NTK regime which eventually helps to prove the global convergence of the algorithm.

More precisely, the training algorithm can be highlighted using three major steps (*Algorithm 1* of [1]): *Phase-1 training*, *Random Perturbation* and *Phase-2 training*. In *Phase-1 training*, the parameters are updated according to typical learning algorithm used in practice (e.g., GD/SGD) which can be represented using following equation:

$$w^{t+1} = w^t - \eta_t \odot g^t; t = 0, 1, 2, ..., \tau - 1 \qquad (4)$$

where, w^t is the learnable parameters at time t, g^t is the updating rule (e.g., for GD/SGD g^t contains the average gradient of the loss function for the training samples/mini-batch with respect to w^t) and η_t is the learning rate. However, *Phase-1 training* is followed by the *Random Perturbation* step at time τ, when gaussian random noise has been added to the weights of all the layers except the last layer as follow:

$$w^{\tau}_{(1:H)} \leftarrow w^{\tau}_{(1:H)} + \delta \qquad (5)$$

where, $w_{(1:H)}$ contains the learnable parameters of all the hidden layers and δ represents the noise vector. The last step in the training process is the *phase-2 training*, where the learning rate $\eta_{t>\tau}$ is modified such that the algorithm enters the NTK regime. Hence, while the *Phase-1 training* ensures that the network is learning representations from the data, the *Random Perturbation* followed by the *Phase-2 training* is utilized to prove the global convergence.

However, to prove the global convergence it is required that the network-dataset combination satisfy the *expressivity condition* stated as follow (*Assumption 1* of [1]):

"*There exists* $w_{(1:H)}$ *such that* $\varphi(w_{(1:H)}) \neq 0$, *where* $\varphi(w_{(1:H)}) := \det([\, h_X^{(H)}(w_{(1:H)}), \mathbf{1}_n]\, [\, h_X^{(H)}(w_{(1:H)}), \mathbf{1}_n]^T)$"

Here, $h_X^{(H)}(w_{(1:H)}) \in \mathbb{R}^{(n \times m_H)}$ is the feature matrix of the last hidden layer, i-th row of which is representing output vector of the last hidden layer for the sample x_i as input and m_H is the number of nodes in the last hidden layer. Essentially, the expressivity condition is ensuring the existence of $w_{(1:H)}$ for which the feature matrix is full-rank. This leads to the confirmation of full-rankness of NTK matrix $\mathbf{K}(w)$ defined in (3). The full-rankness of the NTK matrix at the end of *Phase-1 training*, linear degree of overparameterization at the output layer (i.e., $m_H = \Omega(n)$) and the choice of learning rate at the *Phase-2 training* such that the full-rankness of the NTK matrix is preserved at that phase ensures the global convergence of the proposed algorithm. Unlike general NTK convergence theory, the convergence result hold on data-dependently for this modified algorithm (*Theorem 1 and 3* of [1]). On the other hand, the data dependency of the NTK matrix after the *Phase-1 training* ensures that the network is learning representations from the data. Finally, the linear degree of overparameterization requirement together with the capability of learning representations confirms the global convergence guarantee of the algorithm beyond NTK regime.

The *expressivity condition* is proved to be hold data-independently for fully connected neural networks with softplus activation function ($\sigma(z) = \ln(1 + \exp(\zeta z))/\zeta$) and wide last hidden layer ($m_H \geq n$) (*Theorem 2* of [1]). The condition is verified numerically for ResNet-18 with softplus activation function and wide fully connected layer ($m_H = cn; c = 1.1$) by checking the condition using randomly sampled $w_{(1:H)}$ for different benchmark datasets. Improved test error of ResNet on those datasets compared to baseline algorithm ensures that the modified algorithm does not sacrifice generalization performance while providing global convergence guarantee (*Table 1* of [1]). Also, the modified algorithm reduces training loss further compared to baseline algorithm (*Figure 3* of [1]).

III. Visual Explanations from Deep Neural Networks

Visual explanation of the decision made by a DNN can be obtained by producing a visualization map of the prediction for a particular input. This visualization map will indicate which characteristics of the input influenced the network to make that prediction. For example, for a vision task the visualization map will indicate which pixels of a particular image influenced most to make a particular prediction and what are the features (low, mid or high-level features) learned by different layers of the network to make that prediction.

A. Pixel-space Gradient Visualization

Visualizing CNN predictions in pixel-space is based on calculating the gradient of the class score for the

predicted output class for a particular input image w.r.t. that image [10]. This gradient image highlights those pixels of the image which have most influence to predict the class. In other words, changing the intensities of those pixels of the image will impact most on the prediction score of the class. However, the methods of gradient calculation through relu non-linearity differ in different techniques fall into this category such as in *Backpropagation*, *Deconvolution* and *Guided Backpropagation*.

In *Backpropagation* approach [10], for an input image I_0, the visualization for the predicted class c is obtained by calculating the gradient of the class score S_c which can be denoted by $\frac{\partial S_c}{\partial I_0}$. In other words, the gradient of S_c is backward passed to the input layer.

On the other hand, in *De-convolution* approach [11], a de-convolution network in opposite direction is used to visualize the CNN predictions in pixel-space. It can be inferred that, a deconvolution module which essentially performs convolution with flipped filter of forward direction, replace the convolution module of the forward pass. To reconstruct the un-pooled map in backward pass, a switch is created during forward pass through max-pooling layer which records the maxima of each pooling region and passed it to the deconvolution network for reconstruction.

However, *Backpropagation* and *De-convolution* differ only in the method of backward pass through relu activation [10]. While *Backpropagation* passes the gradients which have positive activation during the forward pass, *De-convolution* passes only the positive gradients (i.e., uses relu in opposite direction). In *Guided Backpropagation* [12], the method of backward pass through relu activation is the intersection of above two methods. In other words, only those positive gradients are backward passed which have positive activation during the forward pass.

As a comparison, *Guided Backpropagation* outperforms the other two in some respects. However, all these methods can produce high-resolution map (i.e., contain fine grained details) with a certain degree.

B. Localization based Visual Explanation

Another line of work to provide visual explanation of predictions from CNN is based on visual localization of the object responsible for the predictions. These visualizations are generated from the activation maps of a convolution layer of the network, usually from the last convolution layer. These visualization maps are referred to as *Class Activation Map* (CAM) in literature.

In [13], a method of generating CAM has been provided for a specific network architecture where the activation maps of the last convolution layer are global average pooled (GAP) and directly connected to the output layer through weights. Then, the CAM of the convolution layer can be generated using following equation:

$$M_c = \sum_k w_k^c A_k \tag{6}$$

where M_c is the CAM for the class c, w_k^c is the weight associated with the class c and the k-th activation map A_k. However, to generate CAM for other network architecture using this method, the portion after the last convolution layer must be replaced by GAP followed by direct connection to the output layer. This requires a retraining of the network.

To produce a more general CAM which can be applicable to any network architecture, a modified approach has been provided in [2] known as *Grad-CAM*, where the gradients of the class score w.r.t. activation maps have been utilized. More precisely, at first the importance factors of activation maps for class c with class score y_c are calculated by global average pooling of the gradient of y_c w.r.t. corresponding activation map A_k as follow:

$$\alpha_k^c = \frac{1}{Z}\sum_{i,j}\left(\frac{\partial y_c}{\partial A_k}\right)_{i,j} \tag{7}$$

where α_k^c is the importance factor, Z is the number of pixels in the activation map and $\frac{\partial y_c}{\partial A_k}$ is the gradient of the class score w.r.t. the activation map A_k. Then, w_k^c in (?) can be replaced by α_k^c to produce rectified class activation map referred to as Grad-CAM as follow:

$$L^c_{\text{Grad-CAM}} = \text{ReLU}\left(\sum_k \alpha_k^c A_k\right) \tag{8}$$

CAM and Grad-CAM both can provide highly class-discriminative map but they lack high-resolution details. On the other hand, approaches in pixel-space gradient visualization can provide high-resolution details but lack class-discriminative ability. In order to utilize the benefits of both of the domains, a combined approach is provided in [2], known as *Guided Grad-CAM*, where the Grad-CAM is upsampled using bilinear interpolation to the input image level and then pointwise multiplied with the visualization generated by guided backpropagation.

Unlike Guided Backpropagation and Grad-CAM which can only produce high-resolution and class-discriminative visualization respectively, Guided Grad CAM can generate both high-resolution and class-discriminative map simultaneously as shown in figure 1. From localization perspective, Grad-CAM outperforms CAM and other weakly-supervised localization methods without sacrificing classification performance (*Table*

(a) Original Image (b) Guided Backpropagation

(c) Guided Grad-CAM (d) Grad-CAM

Fig. 1. Various visualizations of category 'cat' [2]

1 of [2]). From the visualization perspective, human study conducted in [2] suggests that Guided Grad-CAM demonstrates improved class-discrimination ability and reliability compared to its pixel-space visualization counterparts. Moreover, Grad-CAM can analyze failure modes of a network and identify dataset biases. Finally, unlike CAM, due to its generalizability Grade-CAM can provide visual explanation for the prediction of networks deployed in other vision tasks such as image captioning and visual question answering.

IV. Future Research Directions

A. From the Optimization and Generalization Perspective

The modified algorithm presented in [1] provides a new research direction for the convergence and generalization analysis of DNN based on data-depended NTK. However, there are some immediate scopes of improvement of the convergence analysis of the algorithm in terms of precision and convergence results. It can be noted from Theorem 3 of [1] that, there is no estimate of time-complexity (τ) of Phase-1 training. Also, the convergence guarantee depends on the preserveness of full-rank of NTK matrix during Phase-2 training. The full-rankness of NTK is preserved by the suitable choice of learning rate (η_t) at the second phase. But, there is no estimate of η_t given which can preserve the full-rankness. Hence, the analysis results can be made more precise by providing a theoretical estimate of τ and η_t required for convergence. Also, the convergence rate of the algorithm provided is in the sublinear region while convergence rates of GD/SGD in NTK regime found in literature are linear [8]. Hence, improving the convergence rate of the modified algorithm to linear region can be an interesting research direction.

Although, the modified algorithm can provide the representation learning and convergence guarantee simultaneously, the provable generalization guarantee is still unavailable. Hence, one of the major future research directions should be to provide provable generalization guarantee for data depended NTK. Also, providing provable robustness for adversarial training can be considered as an interesting research direction.

B. From the Visual Explanation Perspective

In the generation process of Grad-CAM, the average gradient of the class score w.r.t. activation map has been chosen as the neuron importance factor (α_k^c) but no theoretical support has been provided. Theoretical support for such choices can bring more reliability and interpretability of the visual explanation process and so developing such support can be an interesting research direction. Extending the technique to explain the predictions of the networks deployed in outside the vision tasks (e.g., reinforcement learning, natural language processing, medical diagnosis etc.) can be another future research direction.

V. Conclusion

The modified learning algorithm proposed in [1] can demonstrate representations learning capability and provable global convergence simultaneously. Further, it has been numerically verified that, the proposed learning algorithm does not sacrifice the generalization ability on practical datasets. However, the separation of training process into two phases and the addition of random noise to the parameter after the first phase make it unable to explain the success of GD/SGD in practical overparameterized DNN.

On the other hand, the Grad-CAM proposed in [2] can improve the class-discriminative property and applicable to any CNN architecture. In addition, Guided Grad-Grad CAM can provide high-resolution and class-discriminative map simultaneously. However, some recent papers [14], [15] suggests that, the Grad-CAM does not satisfy the sensitivity axiom [16] and Guided Grad-CAM fails the sanity check of visual explanation methods.

REFERENCES

[1] K. Kawaguchi and Q. Sun, "A recipe for global convergence guarantee in deep neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 9, 2021, pp. 8074–8082.

[2] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 618–626.

[3] Y. Li and Y. Liang, "Learning overparameterized neural networks via stochastic gradient descent on structured data," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 8168–8177.

[4] S. Arora, S. Du, W. Hu, Z. Li, and R. Wang, "Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 322–332.

[5] D. Zou, Y. Cao, D. Zhou, and Q. Gu, "Gradient descent optimizes over-parameterized deep relu networks," *Machine Learning*, vol. 109, no. 3, pp. 467–492, 2020.

[6] Z. Allen-Zhu, Y. Li, and Z. Song, "A convergence theory for deep learning via over-parameterization," in *International Conference on Machine Learning*. PMLR, 2019, pp. 242–252.

[7] Z. Allen-Zhu, Y. Li, and Y. Liang, "Learning and generalization in overparameterized neural networks, going beyond two layers," *Advances in Neural Information Processing Systems*, vol. 32, pp. 6158–6169, 2019.

[8] S. S. Du, X. Zhai, B. Poczos, and A. Singh, "Gradient descent provably optimizes over-parameterized neural networks," in *International Conference on Learning Representations*, 2018.

[9] A. Jacot, F. Gabriel, and C. Hongler, "Neural tangent kernel: convergence and generalization in neural networks," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 8580–8589.

[10] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[11] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*. Springer, 2014, pp. 818–833.

[12] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.

[13] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.

[14] S. Sattarzadeh, M. Sudhakar, K. N. Plataniotis, J. Jang, Y. Jeong, and H. Kim, "Integrated grad-cam: Sensitivity-aware visual explanation of deep convolutional networks via integrated gradient-based scoring," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 1775–1779.

[15] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 9525–9536.

[16] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International Conference on Machine Learning*. PMLR, 2017, pp. 3319–3328.