Deep Regional Gaussian Machine

SunnerLi
National Chiao Tung University
springsunnerli@gmail.com

Abstract

In the structure of deep neural network, the computation of feature transformation can be learned by the training process. Conversely, combining the traditional kernel method into deep learning framework can also obtain the benefit. In this report, we demonstrate another idea which called deep regional gaussian machine. The whole network is composed by the regional gaussian module and the neural network module. In region gaussian module, the numerous but simple kernel machines will compute the similarity map cooperatively. Through the training process, the whole network can learn how to become the combination toward these machines. Additionally, the regional gaussian module can be deployed[1] to any common deep learning model. The deep regional gaussian machine is also tested toward MNIST and CIFAR10 dataset. The results show that the faster convergence phenomenon can be obtained.

1. Introduction & Related work

In the recent years, deep neural network gets the tremendous success in the machine learning territory, especially in computer vision area. For the ILSVRC, some idea of champion model is commonly used in the recent research. The first impressive idea is AlexNet [1]. In the original paper, Alex Krizhevsky claimed that the neural network can learn the abstract feature representation from the low level to high level. [2][3] are also well known to be deployed. In 2015, ResNet [4] is also another big progress in deep learning. Rather than to learn the direct transformation between input image and target, the network can be tent to learn the residual difference between the feature map and the target data. Until 2017, these famous models can learn the object feature with high performance in object recognition tasks.

The kernel method is also a powerful idea in machine learning area. The data can reach the dimensional projection by projecting its feature into high dimensional space. The value of feature in high dimensional space can be calculated by single feature and computed independently or obtain the corresponding value with the other dimension feature mutually. Conversely, the concept of kernel is few used recently since the deep neural network has amount of capacity to learn the appropriate data representation. [5] – [8] shows some previous studies to combine the kernel strategy into deep learning structure. Nevertheless, the kernel computation will be deployed in the whole network which might increase the forward time while the scale of the model is large. Especially, [8] is much closer to our idea, but the idea isn't widely used in the common application and research.

Even though neural network has strong capability to do the feature transformation, the unlimited dimensional feature projection cannot be reached. Conversely, the gaussian kernel can achieve the infinite dimensional transformation. To sum up the advantage of the two approaches, it's potential to find an approach to combine the deep learning framework with kernel method. Moreover, this approach should be also easily adopted and deployed to the application. The final structure not only can absorb the advantage of kernel method with infinite dimensional transformation, but also use the advantage of deep neural network to use the strong flexibility and find the appropriate parameters setting.

2. Methodology

In this section, the fundamental principle of the regional gaussian machine and the structure we purposed will be explained in order.

2.1. Principle

For a typical linear model, the function of the model can be transformed as the dual form, and it is demonstrated in equation 1. In the dual form, the model which is called

[1] The usage of the regional gaussian model (regional gaussian layer) can be found in the repository: https://github.com/SunnerLi/Deep-Regional-Gaussian-Machine

kernel machine compute the linear combination of the kernel and give the prediction. In the equation1, y is the model, x^i and x^j are the independent image. $K(x^i, x^j)$ is the kernel measure and the a is the combination term. The kernel machine can be treated as the linear combination toward the kernel term.

$$\mathrm{y} = \mathrm{K}(x^i, x^j)\mathrm{a} \tag{1}$$

For the traditional kernel concept, the kernel can be regarded as the similarity measure between different data point. We purposed two creative idea in this work. The first is that each pixel point is also independent toward the other pixel point in the same image too. On the other words, not only consider the similarity between each different image, but also measure the similarity between each pixel in the same image.

$$\mathrm{K}(x^i, x^j) = \exp(\textstyle\sum_{k=1}^{M} f_k(|x_k^i - x_k^j|)) \tag{2}$$

The second idea we purposed is the region kernel. In the paper [6], the function of kernel measure is shown in equation 2 which adopt RBF kernel as the computation function. f_k is the underlining function which was purposed in the original paper. Conversely, the computation of the kernel measure is expensive toward the whole features in the image and the extra underlining function should be also considered, we purposed to use another similarity measure which called similarity map by ourselves. The most important is that the similarity map only consider the similarity in the small region. The similarity map (SM) we adopt is written in equation 3.

$$SM\left(x_j^{(i)}\right) = \sqrt[k]{\textstyle\prod_{m\in\xi} \exp(-\left\|x_j^{(i)} - x_m^{(i)}\right\|^2)} \tag{3}$$

In the equation 3, the $SM\left(x_j^{(i)}\right)$ represents the similarity map value toward the *j* pixel in *i* image, k is the region size, ξ is the index set of the neighbor whose center is the *j* pixel, and *m* is the specific index in the index set. The goal of this measure is to calculate the geometric average of gaussian kernel among the neighbor pixels and the center pixel.

The Figure 1 demonstrates the idea of regional kernel machine with regional gaussian kernel. The pair of each circle and square region is a simple machine which only calculate the similarity within a limited region. The colorful circle represents the combination term and the rendered square region represents the kernel. Through the training, the combination term should be obtained automatically.

Compare to the convolution operation, the sliding window mechanism is adopted by both operations. The most different is the function of the window. In the convolution operation, the filter is used to do the convolution operation toward the window, and the filter contains weight which will participate the computation process. Conversely, the kernel is used to compute the geometric average of the gaussian similarity toward the window, and the filter doesn't contain the weight but just limit the area which do the computation in.

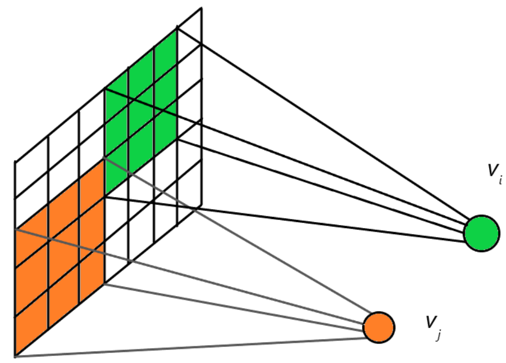

Figure 1. The idea of kernel machine with regional gaussian kernel.

In our idea, the whole network can be regard as being composed by bunch of simple kernel machines. Each kernel machine only learns the regional similarity in the small area. Additionally, the whole structure should merge the result of each machine and give the final prediction. The character of the network is not only learning the combination term in the single kernel machine, but also realizing the combination of the numerous kernel machines and cooperating to learn the boundary for the target task.

2.2. The structure

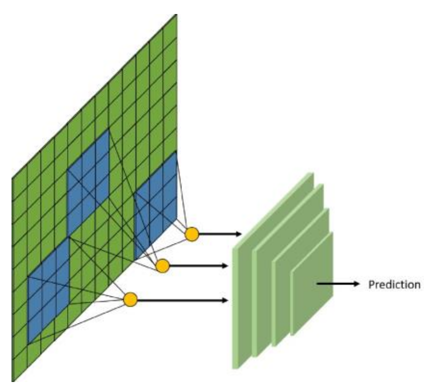

Figure 2. The structure of deep regional gaussian machine.

The Figure 2 illustrates the structure of deep regional gaussian machine. The whole model can be separated as two parts. The first part is regional gaussian module which contains numerous kernel machines. The bunch of kernel machines will conduct toward the input image, and the corresponding similarity map will be generated. The second part is the neural network module which accepts the similarity map from the regional gaussian module. Furthermore, the non-linear computation will be conducted and the neural network module will gives the final prediction. In this structure, the network is treated as the combination term toward the given similarity map.

$$y = f(K(x^i, x^j)|\, a) \tag{4}$$

By the concept of being combination, the function of whole deep regional gaussian machine can be written as the equation 4. The f is the neural network which represent a non-linear combination term. The neural network learns the combination a in each single kernel machine and learn the cooperation between the numerous of kernel machines.

By this design, the efficiency advantage can also be obtained. In the paper [7], the goal of convolution is to calculate the filter of the convolution layer and the network treats the filter as the kernel in the kernel concept, and each convolution layer should adopt the computation which might spend a lot of time to forward. Oppositely, the computation of similarity map is only deployed in the first layer of the network, the higher proceeding efficiency can be achieved.

3. Experiment Result

In this section, the result between the deep regional gaussian machine and the traditional neural network will be described. For the whole experiments, we use the server whose CPU is Intel Core i7-4790. The GPU acceleration is also adopted which is GeForce GTX 1070. To show the quick convergence advantage of our model, the epoch is only set as 10.

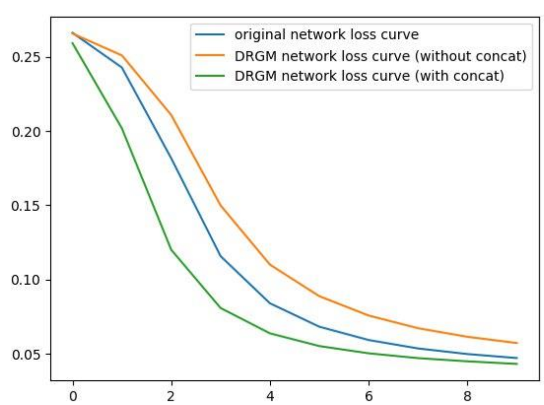

Figure 3. The loss curve of experiment 1 which test on MNIST dataset.

In the first experiment, the critical issue is if the original pixel intensity image should be discarded while the network only remains the similarity map to do the prediction. The Figure 3 shows the results. The orange line is the loss curve of the deep regional gaussian machine which only use similarity map to do the computation; the blue curve is the loss trend of the usual deep neural network, and the green line is the loss curve of the deep regional gaussian machine which use both pixel intensity image and similarity map to do the prediction.

For the testing accuracy, the deep neural network can achieve 89.53% accuracy; the deep regional gaussian machine with remaining pixel intensity image can obtain 89.73% accuracy. The results show that it's also important to remain the pixel intensity image to reach the higher performance.

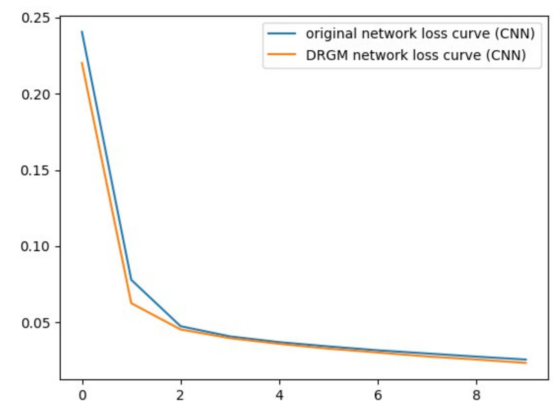

Figure 4. The loss curve of experiment 2 which test on MNIST dataset.

In the second experiment, the concept of regional gaussian machine can be also deployed to the convolutional neural network structure. Rather than the paper [8], our design can be more easily to implement on the common popular model. The experiment result is illustrated in Figure 4. The blue line is the loss curve of usual convolutional neural network, and the orange line is the loss curve of deep regional gaussian machine. For the testing performance, the deep neural network obtains for 94.28% accuracy while the deep regional gaussian machine can achieve for 94.5% accuracy. The result shows that the deep regional gaussian machine can also achieve more quick convergence during the training.

How to determine the region size is also a critical issue for the model. In the experiment 3, three different region sizes are considered, including 1, 3 and 5. We also consider the size is 1 which is called self-gaussian and this design doesn't consider any similarity to the neighbor but only remain the similarity to itself. The Figure 5 illustrates the loss curve of the experiment. The blue curve is the loss curve of usual convolutional neural network; the orange curve is the loss curve of self-gaussian; the green curve is the loss curve whose region size is 3, and the red curve is the case whose region size is 5.

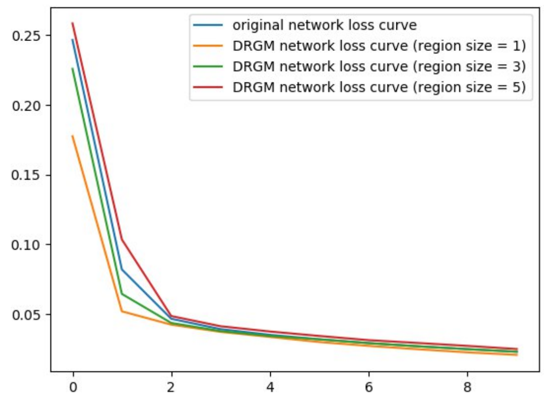

Figure 5. The loss curve of experiment 3 which test on MNIST dataset.

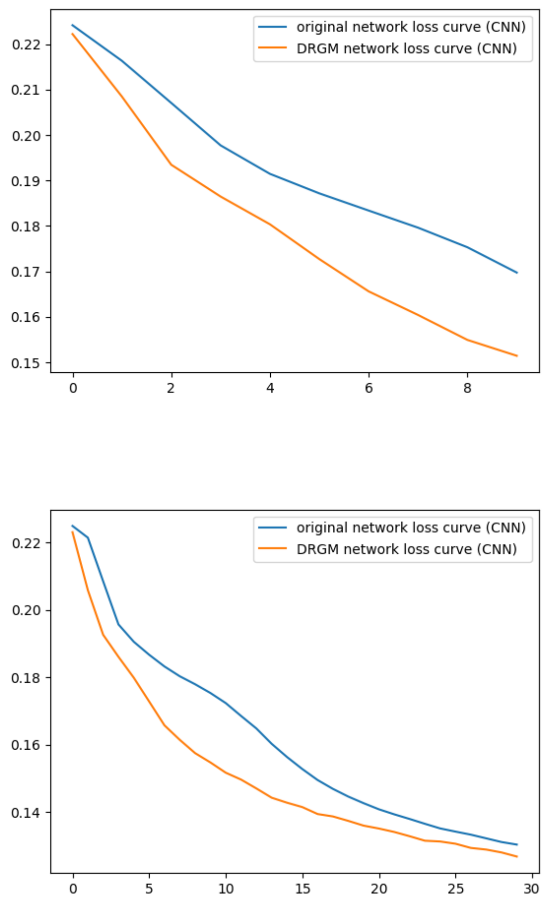

Figure 6. The loss curve of final experiment which test on CIFAR-10 dataset

The result shows that the region size 1 and 3 can both achieve faster convergence. This result can provide the user chance to choose which case should be adopted. If the user tries to train for more deep structure in the case of shortage computation resource, the self-gaussian might be more efficient. On the other hand, the region size of 3 can also be considered in the enough computation environment since it also considers the similarity among the neighbor pixels.

In the last experiment, we want to validate if the deep regional gaussian machine can also work for the real-image task. Consequently, the CIFAR-10 dataset is also tested. The Figure 6 demonstrates the result. The blue curve is the loss curve of usual convolutional neural network and the orange curve is the loss curve of deep regional gaussian machine with self-gaussian. The upper sub-image shows the experiment which training epoch is 10, and the training epoch of lower sub-image is 30. The results show that more considering the similarity map can also achieve more quick convergence.

4. Conclusion

This report shows the idea which called deep regional gaussian machine. The whole structure is composed by two parts: regional gaussian module and neural network module. In the regional gaussian module, the region gaussian computation process is adopted to obtain the similarity map. The similarity map will concatenate the original intensity pixel image and be sent into the neural network module. The neural network module will receive the result which is sent from regional gaussian module and compute the final prediction. The idea of regional kernel can achieve more quick convergence during training. Additionally, we also purpose self-gaussian idea which can reach the lower loss value but the same computation efficiency during network forwarding. By the series of experiments, this design can obtain the advantage toward the artificial dataset and the natural image dataset.

Easily to deploy to the various models is the most advantage of the regional gaussian idea. We merge the whole kernel computation process into a single layer that the user can just use this idea into their own model. Through this design, we wish that the regional kernel idea can bring more higher performance in various tasks.

Reference

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems (NIPS)*, USA, 3-8, December 2012, pp. 1097-1105.

[2] Karen Simonyan and Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv:1409.1556 [cs.CV], September 2014.

[3] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going Deeper with Convolutions," in *Computer Vision and Pattern Recognition (CVPR)*, Boston, USA, 8-10, June 2015, pp. 1-9.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep Residual Learning for Image Recognition," in *Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, 26 June – 1 July 2016, pp. 770-778.

[5] Youngmin Cho and Lawrence K. Saul, "kernel methods for deep learning," in Advances in *Neural Information Processing Systems (NIPS)*, Vancouver, Canada, 6-11 December 2009, pp. 1-9.

[6] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P. Xing, "Deep Kernel Learning," arXiv:1511.02222 [cs.LG], November 2015.

[7] Julien Mairal, Piotr Koniusz, Zaid Harchaoui, and Cordelia Schmid, "Convolutional Kernel Networks," in *Neural Information Processing Systems (NIPS)*, Montréal, Canada, 8-13, December 2014, pp. 1-9

[8] Linh Le, Jie Hao, Ying Xie, and Jennifer Priestley, "Deep Kernel: Learning Kernel Function from Data Using Deep Neural Network," in *International Conference on Big Data Computing Applications and Technologies (BDCAT)*, Shanghai, China, 6-9, December 2016, pp. 1-7.