Neural networks with fixed binary random projections improve accuracy in classifying noisy data

Zijin Yang[1], Achim Schilling[2], Andreas Maier[1], Patrick Krauss[2]

[1]Pattern Recognition Lab, FAU Erlangen-Nürnberg
[2]Neuroscience Lab, University Hospital Erlangen
michael.yang@fau.de

Abstract. The trend of Artificial Neural Networks becoming "bigger" and "deeper" persists. Training these networks using back-propagation is considered biologically implausible and a time-consuming task. Hence, we investigate how far we can go with fixed binary random projections (BRPs), an approach which reduces the number of trainable parameters using localized receptive fields and binary weights. Evaluating this approach on the MNIST dataset we discovered that contrary to models with fully-trained dense weights, models using fixed localized sparse BRPs yield equally good performance in terms of accuracy, saving 98% computations when generating the hidden representation for the input. Furthermore, we discovered that using BRPs leads to a more robust performance – up to 56% better compared to dense models – in terms of classifying noisy inputs.

1 Introduction

In recent years, algorithms of artificial intelligence (AI) and machine learning (ML) have undergone significant improvements, enabling machines to automatically perform tasks such as semantic analysis of images or language patterns with high accuracy and reliability. To increase the performance of artificial neural networks (ANN), which are a major part of this development, the model architecture tends to become more complex by using a large number of parameters and hidden layers [1]. These "deep" architectures allow for a broad spectrum of different network topologies with various numbers of hidden neurons, hidden layers, parameters configurations, or optimization strategies. As a result, it is increasingly hard to train these networks and to find a suitable hyper-parameter configuration for the task at hand.

As we all have experienced, the human visual system can effortlessly make sense of a novel scene even if the salient patterns are heavily altered (e.g. recognizing humans by their reflections in a strongly distorted mirror). This is possible as the human visual system is capable to generalize across different levels of image manipulations such as contrast reduction, additive noise, or novel eidolon-distortions. However, ANN algorithms are typically very sensitive to

these kind of image characteristics. Consider following example: A properly configured ResNet-152 [2] trained on standard color images and tested on a similar test distribution performs close to or even surpasses human observer performance [3]. Introducing the same type of noise in both training and testing datasets does not interfere with the performance of the network. However, if the network is trained on images with e.g. salt-and-pepper noise and tested on images with e.g. uniform noise, the performance is at chance level, even though both noise types do not seem much different to human observers [3]. Since such changes deep in the human brain are poorly understood, drawing inspirations from artificial neural networks has become a way to "unlock" the world of organisms [4].

Dasgupta et al. have presented a neural algorithm derived from the biological structure of fruit flies' olfactory system [5]. The fruit fly's brain has an elegant and efficient way of performing similarity searches. Unlike the common locality-sensitive hashing methods, which reduce the dimension of the input item and assign short "hashes" to each item such that similar items are more likely to be assigned to same or similar hash tag, the fruit fly brain expands the dimension. The brain then stores only 5 % of the expanded neurons with top activity as the hash tag for that odor [5]. Only a limited number of hash length – the percentage of the neurons with top activity that stored by the system – was investigated and the relationship between the length of the hash and the size of the projections is still unclear.

The main goal of this work is to apply this concept in the form of fixed BRPs [5] and localized connectivity [6] to neural networks. With this we hope to reduce training time while preserving the same performance and having a more generalized model. To investigate whether this approach can mimic the process of biological visual systems, experiments are performed by training the models on clear images and testing on noisy images.

2 Material and methods

2.1 Model

An illustration of the basic concept of this work is depicted in Fig. 1. The first step is to normalize the input image by converting the pixel intensities from range $[0, 255]$ to range $[0, 1]$. Z-score normalization is not necessary because all the pixels are in the same intensity range i. e. the values have equal contribution. The rescaling of the input is not necessary but encouraged to improve time efficiency.

The second step is to project the inputs to the hidden layer using sparse, binary random weights (either one or zero) with localized receptive fields as can be seen in Fig. 1. Theoretically, if the number of hidden neurons is infinite, we can have an infinite large amount of different filter kernels (localized receptive fields) to project the patches from the input to the hidden layers. Thus, we can have activations for all possible patterns. A fruit fly projects 50 projection neurons (PNs) to 2000 Kenyon cells (KCs), which is a 40-fold expansion of neurons [5].

Likewise, the number of hidden neurons is a multiple-fold expansion of the input neurons which contains 784 neurons, due to the image size of 28×28.

The third step is the Winner-Takes-All procedure (Fig. 1, 3rd step). A fraction i. e. $a\%$ of the highest-activating neurons remain activated and the rest of the neurons are inhibited i. e. set to zero. a here is the hash length which indicates the number of the activated neurons. With the increase in neurons in the hidden layer, not all neurons contain meaningful information. Since the filters are generated in a random manner, those "nonsense" filters, which are not correlated to the actual pattern of the input, would generate low values in the hidden layer. Based on the choice of hash-length a, the Winner-Takes-All procedure can assign different input into different "Tags", which are better separable in higher-dimensional space.

The final step is to train a classifier i. e. a fully connected layer as a read-out layer for the generated "Tags" to actually classify (Fig. 1, 4th step). Weights for this layer are initialized using He et al. initialization [7] and are trained using back-propagation.

2.2 Dataset

To further investigate the capability of BRPs, MNIST images with additive Gaussian noise are generated for test purpose. Additive Gaussian noise is gen-

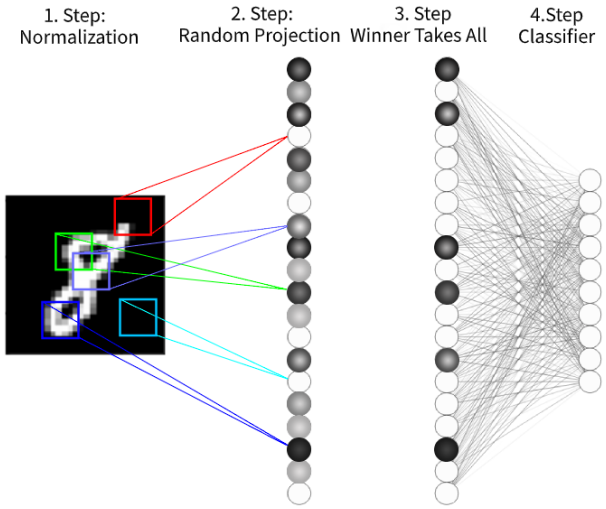

Fig. 1. An illustration of the basic neural network. Left column: an input image from the MNIST data set with five localized receptive fields selected by randomly chosen pixel center, respectively. Middle left column: the hidden layer of the basic network. It consists of hidden neurons with different intensities. Middle right column: the same hidden layer but only 20% of the neurons with higher activations are still activating and the remaining 80% are inhibited. Right column: the output layer with 10 output neurons, indicating 10 target classes. Darker circles indicate higher activations while white circles indicate inhibition.

erated by randomly drawing numbers from a Gaussian distribution as follow to avoid negative values:

$$\mathcal{N}'_{i,j} = I \cdot \frac{\mathcal{N}^*_{i,j}}{2} \tag{1}$$

$$\mathcal{N}^*_{i,j} \sim \mathcal{N}(2,1) \tag{2}$$

where $\mathcal{N}'_{i,j}$ denotes the noise added to the original image at position (i,j), I denotes the noise intensity, $\mathcal{N}^*_{i,j}$ is the basic noise value generated by a Gaussian distribution. Test results are shown in Fig.3 where all models are trained on 12000 original MNIST images and tested on images with different noise intensities. Each noise intensity contains 2000 images with the same level of additive Gaussian noise.

3 Results

As shown in Fig. 2, models with proper parameter settings using fixed BRPs yield almost the same level of performance as a fully-connected and fully trained

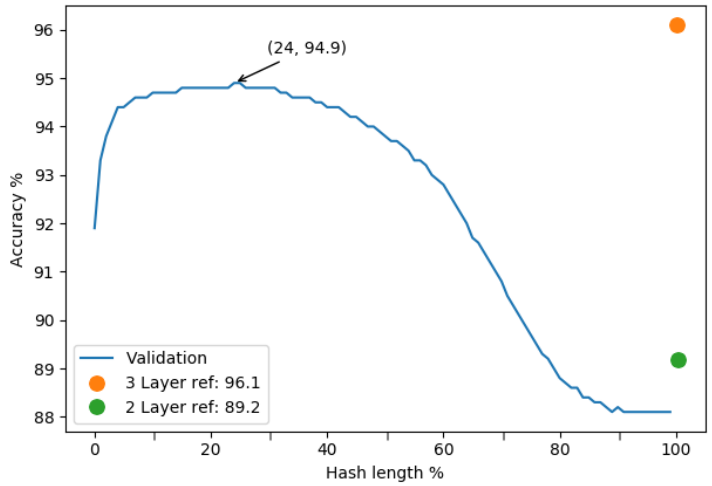

Fig. 2. Average validation accuracy for a model with a single hidden layer with hidden layer size $n_h = 5000$, dependent on hash length (%) over 50 trials, trained and tested on MNIST without noise. Parameters: drop rate $d = 0.9$, batch size $b = 32$, *Adam* optimizer, learning rate $lr = 0.001$, patch size $p = (10, 10)$, hidden layer size $n_h = 5000$. Each model is taken from the best model during 200 epochs. Orange dot(top one) and green dot(bottom one) on the right side of the figure indicate 3 layer reference i. e fully connected and fully trained single hidden layer network with the same hidden layer size $n_h = 5000$, and 2 layer reference i. e a fully trained network directly propagate from input layer to output layer with no hidden layer, respectively.

network with the same hidden layer size. The curve has shown that a hash length in the range of $(5\%, 30\%)$ leads to similar high-performance (at around 95%). As the hash length increases, the performance first slowly decreases (from 35% to 60%), then dramatically decreases(from 60% to 85%), and then remains relatively constant (around 88%).

As can be seen in Fig. 3, models using fixed BRPs start at almost the same accuracy compared to the reference model where each layer is fully connected with the next layer and fully trained. As the noise intensity increases, the difference between the performance in terms of accuracy of both BRPs and Ref models becomes more apparent. All results suggest that using BRPs leads to more robust performance in terms of classifying noisy images.

For the cost of the computations, 90% of the weights between the localized input layer and hidden layer are zero. The number of multiplication operations here with localized receptive fields are reduced by 98%. The remaining 2% of the weights are all ones, therefore a floating point matrix multiplication is "con-

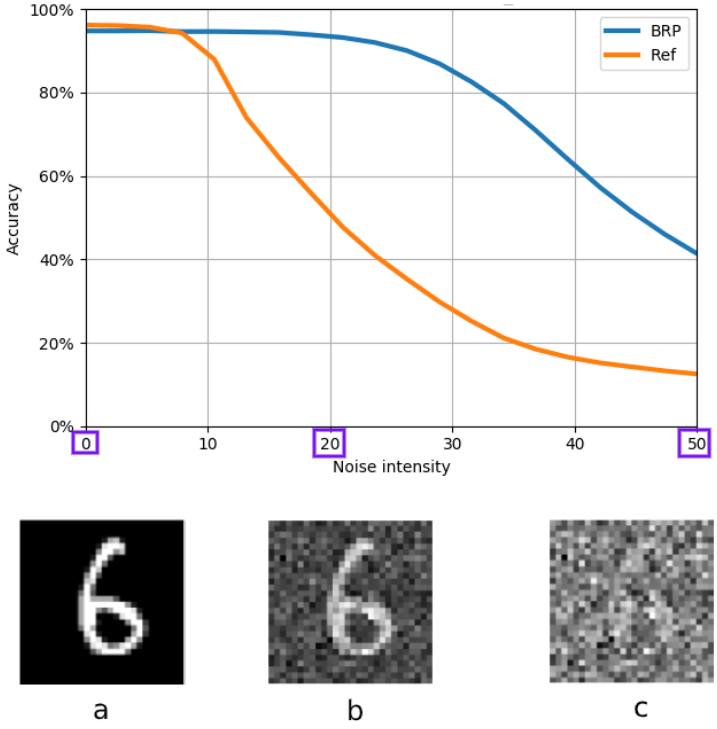

Fig. 3. Test accuracy on noisy images. BRP: model with a single hidden layer with BRPs where only the weights from the readout layer are trained. Ref: a fully-connected and fully trained reference model with the same architecture. Hidden layer size $n_h = 5000$. Image a, b and c: image with zero noise intensity, a noise intensity of 20 and a noisy intensity of 50, respectively. Training parameters are described as in the caption of Fig 2

verted" to accumulations, which further reduce the computational cost. Furthermore, the parameters needed to be trained are greatly reduced by fixation of the weights between input layer and hidden layer, leading to a reduction of the training time.

4 Discussion

In this work we analyze the impact of the use of binary random projections on either the noise robustness and efficiency of neural networks. This analysis leads to the following conclusions: an artificial neural network does not need to be fully trained. Networks using localized receptive fields and fixed BRPs perform equally well compared to networks with all-to-all trained connectivity. This method helps to reduce a large amount of trainable parameters and training time while preserving the performance on the task and improving the ability of being noise-invariant, allowing memory and computational benefits.

As certain steps have not be covered in the frame for this work, the following denotes suggestions for further work on this topic. It is suggested to further test the performance of BRPs for convolutional neural networks. With enough differently initialized kernels, it is possible to achieve the same performance as a fully trained model. Furthermore, as investigated by Hoffer et al. a classifier can be fixed with little to none loss of accuracy for most tasks [8]. The combination of BRPs and a fixed classifier might result in a model that does not need to be trained at all, reducing training time and trainable parameters to zero. Other type of noise e. g Impulse noise, Poisson noise are still not investigated. For further understanding of the method's behaviour when classifying different type of noisy images, it is suggested to perform more tests with more complicated architectures.3075

References

1. Chakraborty B, Shaw B, Aich J, et al. Does deeper network lead to better accuracy: a case study on handwritten Devanagari characters. Proc Int Anal Doc Syst (DAS). 2018 April; p. 411–417.
2. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit; 2016. p. 770–778.
3. Geirhos R, Temme CRM, Rauber J, et al. Generalisation in humans and deep neural networks. In: Adv Neural Inf Process Syst. Curran Associates; 2018. p. 7538–7550.
4. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015 May;521:436–444.
5. Dasgupta S, Stevens CF, Navlakha S. A neural algorithm for a fundamental computing problem. Science. 2017;358(6364):793–796.
6. Illing B, Gerstner W, Brea J. Biologically plausible deep learning - But how far can we go with shallow networks? Neural Networks. 2019;118:90–101.
7. He K, Zhang X, Ren S, et al. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: Proc IEEE Int Conf Comput Vis; 2015. p. 1026–1034.
8. Hoffer E, Hubara I, Soudry D. Fix your classifier: the marginal value of training the last weight layer. In: Proc Conf Learn Represent; 2018. .