International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-1, October, 2019

Visual Modeling of Data using Convolutional Neural Networks

Moulana Mohammed, M. Venkata Sai Sowmya, Y. Akhila, B. Naga Megana

Abstract**: *Artificial Intelligence has been showing monumental growth in filling the gap between the capabilities of humans and machines. Researchers and scientists work on many aspects to make new things happen. Computer Vision is one of them. To make the system to visualize, neural networks are used. Some of the well-known Neural Networks include CNN, Feedforward Neural Networks (FNN), and Recurrent Neural Networks (RNN) and so on. Among them, CNN is the correct choice for computer vision because they learn relevant features from an image or video similar to the human brain. In this paper, the dataset used is CIFAR-10 (Canadian Institute for Advanced Research) which contains 60,000 images in the size of 32x32. Those images are divided into 10 different classes which contains both training and testing images. The training images are 50,000 and testing images are 10,000. The ten different classes contain airplanes, automobiles, birds, cat, ship, truck, deer, dog, frog and horse images. This paper was mainly concentrated on improving performance using normalization layers and comparing the accuracy achieved using different activation functions like ReLU and Tanh.***

Keywords: CNN, Computer Vision, Normalization layer.

I. INTRODUCTION

In deep learning, batch normalization made the deep CNN efficient. This paper concentrates on designing a CNN architecture which performs segmentation by extracting fine grained features in the images. The network is trained on 60,000 images with no scatter correction. It also includes pre-processing stages for reducing the noise in the images by using normalization. In this work, one hot encoding is the technique used for the categorization of images. In this work, the technique used for the categorization of images is One hot encoding. In this process, the variables that can be categorized are converted into a different form and are provided to Machine Algorithms to perform well in prediction. It also categorizes the images into their respective groups. CIFAR-10 dataset is a collection of images that can be used to teach a computer how to recognize objects. As the images are of low resolution, it became quick to try different activation functions on the dataset. The activation functions used here are ReLU (Rectified Linear Units) and Tanh (Tangent Hyperbolic function). ReLU function returns a value of zero for negative inputs and the same face value for the positive inputs i.e.., ReLU(x) = max(0, x). Tanh activation function also known as Hyperbolic tangent has its range from (-1, 1). It is used for classification between two classes. The network is further sampled by Max Pooling. It performs the spatial down technique of the feature map and retains only the most relevant information. The paper is organized as follows:

In section II, the existing literature is analyzed. In Section III a method is proposed and a flowchart. In section IV, a comparison of the results is projected using bar graphs. In Sections V, the conclusion and in Section VI, the references are included.

The below mentioned Images Fig. 1.1, Fig. 1.2 are useful for this paper to compare the activation functions and Fig. 1.3 to understand CNN methodology.

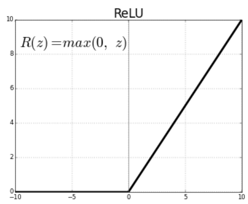

Fig. 1.1 Courtesy: [11]

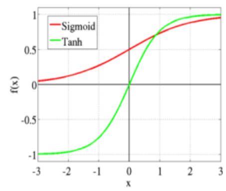

Fig. 1.2 Courtesy: [12]

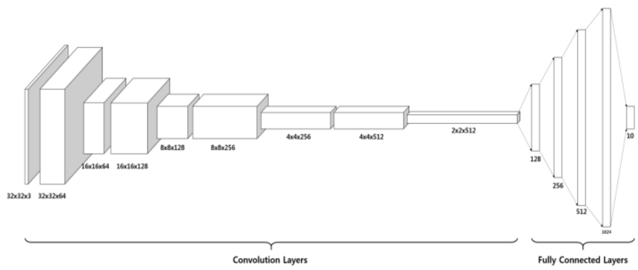

Fig. 1.3 Courtesy: [13]

II. LITERATURE REVIEW

Vignesh Thakkar [3] et.al., explained how state-of-art CNN is performed by using Dense net, VGG and some other models of Neural Network with CIFAR-10 dataset as input to these models. They described the importance of Batch Normalization in improving the performance of neural networks and its ability to train deep convolutional networks. The results produced by applying different activation functions to these models were also recorded and produced.

Revised Manuscript Received on October 30, 2019.

* Correspondence Author
Moulana Mohammed*, CSE Department, Koneru Lakshmaiah Educational Foundation, Guntur, India. Email: moulanaphd@gmail.com
M. Sowmya, CSE Department, Koneru Lakshmaiah Educational Foundation, Guntur, India. Email: sowmyamudunuri15@gmail.com
Y. Akhila, CSE Department, Koneru Lakshmaiah Educational Foundation, Guntur, India. Email: yarradlaakhilareddy@gmail.com
B. Naga Megana, CSE Department, Koneru Lakshmaiah Educational Foundation, Guntur, India. Email: nmeganabilla@gmail.com

Tinyi Liu[9] et.al., provided an analysis on the CNN algorithm for both forward and backward propagations. They also described how a particular CNN is used to implement face recognition problem.
Rasim Caner Calik [5] et.al., mainly focused on using embedded systems for performing Image classification. The embedded systems usually consist of a limited amount of memory. The framework produced in this paper used a little amount of memory and achieved very good classification accuracy. Their experimental results showed that the time taken for classifying the images is also less which is nearly 0.77ms.
Tianmei Guo [2] et.al., explained how they built a simple CNN for Image classification. They used standard datasets like MNIST and CIFAR-10 for implementing the same. They also analyzed different optimization algorithms for solving the optimal parameters that influence image classification.
Raveen Doon [7] et.al., provided information on CIFAR-10 (Canadian Institute For Advanced Research) dataset and also gave the architecture of CNN. They calculated and visualized the accuracy using the plots. The network achieved 90% of training accuracy and 87.57% of testing accuracy.
Joseph Bullock [1] et.al., classified X-ray images using CNN. The dataset they used for classification contains 150 X-Ray images. This dataset was augmented to produce 7000 images randomly. The network they built was able to produce considerably good results.
Alex Krizhevsky [6] et.al., described the CIFAR-10 dataset and also how the convolutional neural network is trained. They used a few tricks to avoid overfitting and different activation functions for training the DBN.
Tayeb Alipourfard [10] et.al., provided a framework which combined CNN and the subspace reduction method for preparing informative methods. The network they build produced an exceptional classification performance for limited training samples.
Neha Sharma [4] et.al., focussed on analyzing the performance of three popular networks namely AlexNet, GoogleNet and ResNet50. The datasets used for their study are ImageNet, CIFAR-10 and CIFAR-100. The results showed that GoogleNet and ResNet50 have comparatively better accuracy than AlexNet.
Sajid Nazir [8] et.al., tells us how hyperparameters play a major role in the convolutional neural networks (CNN). Based on these hyperparameters values the network speed and accuracy will vary.

III. PROPOSED APPROACH

Benchmark dataset CIFAR-10

The dataset used for our model is CIFAR-10 which contains color images of size 32x32. They belong to 10 different classes represented in Fig. 3.1.

Fig. 3.1 Courtesy: [2]

Convolution Neural Network Layers

Fig. 3.2 represents a basic CNN which is used to classify images. The network will accept an input image which looks like an array of pixels, processes it and classifies it into a category. The convolution neural network usually consists of the Convolution layer, Pooling Layer, Dense layer, Normalization layer.

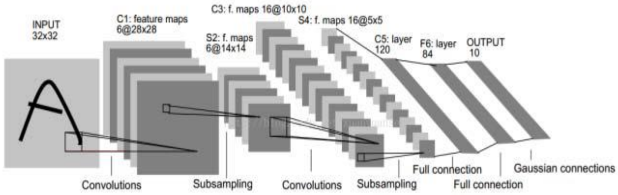

Fig. 3.2 Courtesy: [2]

Convolution Layer is the fundamental layer of the network. The feature representations of inputs are learnt by this layer. The layer performs two processes namely convolution process and sampling process. A convolution process represented in Fig. 3.3 will compute the dot product of the weights and the connected regions in the input. The change in dimensions will depend upon the filter size and no. of filters used.

- An image matrix (volume) of dimension **(h x w x d)**
- A filter **(f_h x f_w x d)**
- Outputs a volume dimension **(h - f_h + 1) x (w - f_w + 1) x 1**

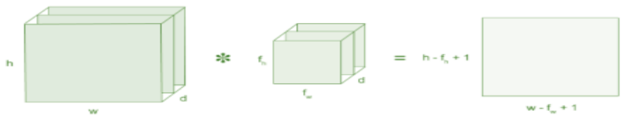

Fig. 3.3 Convolutional Layer

Pooling Layer's main aim is to reduce the number of spatial dimensions explained in Fig. 3.4. The different types of pooling functions available are max, average and sum. Max pooling is most commonly used in many networks. In max pooling, the maximum element from the window will be considered.

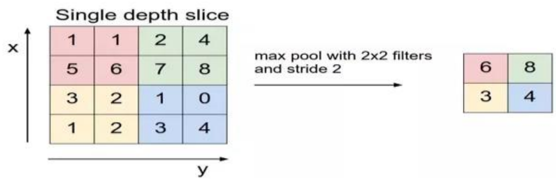

Fig. 3.4 pooling Layer

Dense Layer/ Fully Connected Layer focuses on dimensionality reduction and at the output layer, it will reach a classification decision. The first fully connected layer will predict the correct label by applying weights to the outputs derived from feature analysis. The connection between one layer to another layer is given below in Fig. 3.5.

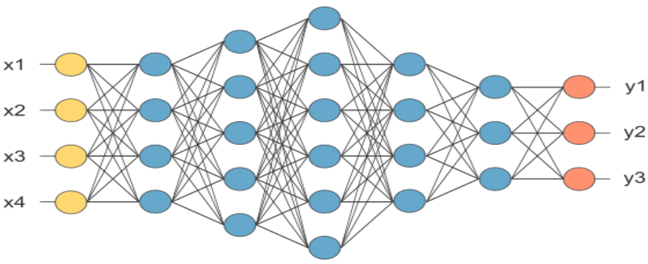

Fig. 3.5 Fully Connected Layer

Batch Normalization Layer will speed up the training of CNN and reduces the sensitivity to network initialization. This layer will also allow high learning rates and reduces the number of epochs required for training deep networks.

In this paper, CNN is used to visually analyze the images present in CIFAR-10 dataset. The below-mentioned steps are followed to process the dataset in every layer of CNN. The numbers of fitting neurons are calculated based on the following equation (1).

$$N = \frac{W-F+2P}{S} + 1 \tag{1}$$

Where

W is the size of the input,
F is the receptive field size,
P is the padding and
S is the stride.

The Proposed Algorithm is as follows:

Step 1: *Load the CIFAR-10 dataset and send it to the input layers.*
Step 2: *Pre-process the data*
Step 3: *For each layer of CNN*
- ***Step 3.1:*** *Fit the number of Neurons by using the number of fitting neurons formula.*
- ***Step 3.2:*** *Use the activation functions* ***Tanh*** *and* ***ReLU.***
- ***Step 3.3:*** *Apply Max Pooling.*
- ***Step 3.4:*** *Normalize the output of Pooling.*
- ***Step 3.5:*** *Use the Dropout layer to reduce neurons.*
- ***Step 3.6:*** *Normalize the Dropout output.*
- ***Step 3.7:*** *Minimize the loss function.*

Step 4: *Repeat the above steps until the number of layers.*
Step 5*: Calculate the accuracy.*

The entire model consists of 14 layers mentioned in Fig. 3.6.

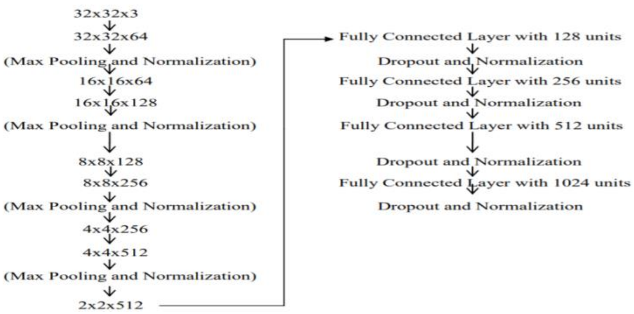

Fig. 3.6 Flow Diagram for the above methodology

A) FLOW CHART

During the representation of the flow of the process, the CIFAR-10 dataset will be pre-processed to remove noisy data by using the normalization. Then the built model takes the pre-processed CIFAR-10 dataset as input and sends it to the convolution layers for performing the convolution process. Activation functions such as ReLU and Tanh are applied for the outputs generated from convolutional layers to convert the non-linear outputs into linear outputs. Then, to reduce the dimensionality of image data we included a Max Pooling layer which is followed by a Normalization layer. To further enhance the performance of the network a Dropout layer is included which will again be followed by a normalization layer. The Loss function will be calculated and minimized using Adam optimizer. Finally, we will calculate the accuracy of the model when there are no convolutional layers available in the network. The diagrammatic representation is shown in Fig. 3.7.

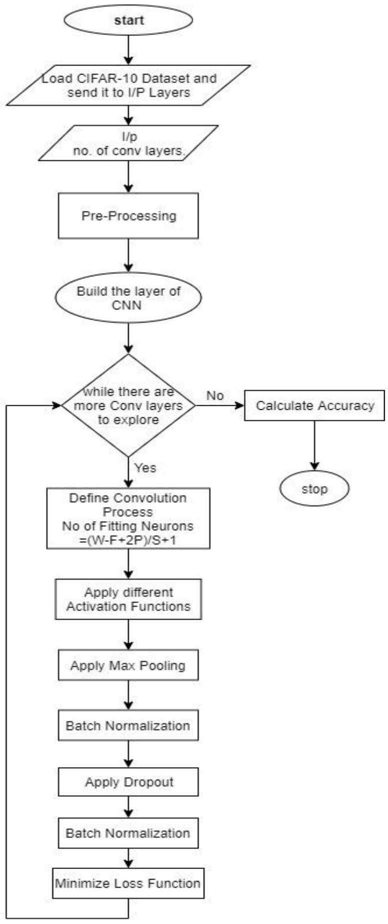

Fig. 3.7 Flow Diagram for the CIFAR-10 using CNN methodology

IV. RESULTS

The batch normalization layer is optional. However, the inclusion of this layer will yield improved performance. When compared to other existing models, our model doesn't have the best accuracy rate but the network structure is simple and reliable. The constraints on memory and parameters are also less. In the Table- I, the loss values produced by the activation functions ReLU and Tanh are recorded.

Table- I: Loss values for Tanh and ReLU activation functions

Epoch	Loss from Tanh function	Loss from ReLU function
1	2.3138	2.2432
1	2.2963	1.9785
1	2.2949	1.5493
1	2.2945	1.7080
1	2.2918	1.5803
2	2.2869	1.4503
2	2.2854	1.1928
2	2.2719	0.9787

2	2.2691	1.1011
2	2.2631	1.0058
3	0.7088	0.9216
3	0.5986	0.7217
3	0.3978	0.5784
3	0.3376	0.6975
3	0.3225	0.5840
4	0.2835	0.6615
4	0.2565	0.4138
4	0.2351	0.2725
4	0.2254	0.4141
4	0.1819	0.2730
5	0.1723	0.3814
5	0.1614	0.2562
5	0.1532	0.1421
5	0.1519	0.1992
5	0.1515	0.1337

From Table- I it is clear that the loss values from the Tanh activation function are more when compared to the loss values obtained when the ReLU activation function is used. It is represented in Fig. 4.1.

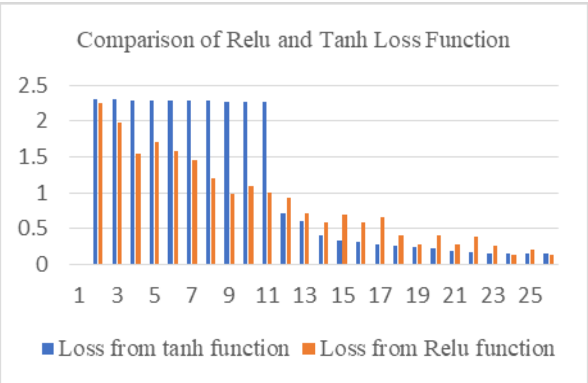

Fig. 4.1: Bar plot comparison of ReLU and Tanh Loss Function

In the Table-II, the validation accuracy values produced by the activation functions ReLU and Tanh are recorded.

Table- II: Validation Accuracy values for Tanh and ReLU activation functions

Epoch	Validation Accuracy for Tanh Function	Validation Accuracy for ReLU function
1	0.0998	0.1606
1	0.0942	0.1796
1	0.0942	0.2524
1	0.105	0.3174
1	0.0998	0.381
2	0.0998	0.4556
2	0.0978	0.4854
2	0.0942	0.5272
2	0.105	0.5702
2	0.105	0.575
3	0.5774	0.605
3	0.5966	0.6412
3	0.6156	0.6284
3	0.6106	0.6674
3	0.6164	0.6676
4	0.6204	0.6634
4	0.6232	0.688
4	0.6434	0.6896
4	0.6326	0.6994
4	0.6572	0.7184
5	0.6706	0.6916
5	0.6813	0.7094
5	0.6858	0.6714
5	0.7106	0.729
5	0.7108	0.7338

By observing the Table- II, the validation accuracy is more when the activation function is ReLU compared to the results achieved when the activation function is Tanh and the test accuracy for ReLU and Tanh is 0.731289 and 0.701035 respectively. So, the best choice in this evaluation process for the activation function is ReLU represented in Fig. 4.2.

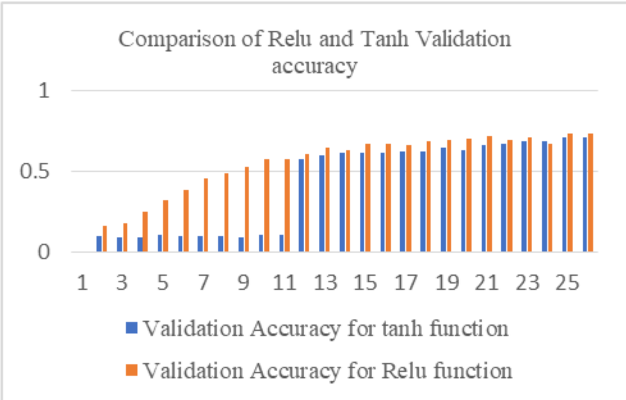

Fig. 4.2: Bar plot comparison of ReLU and Tanh Validation accuracy

V. CONCLUSION

We have analyzed the CIFAR-10 dataset using a four-layered CNN architecture. The accuracy we achieved is nearly 75%. This model included a batch normalization layer which is not used in most of the other network models that we had studied in the literature survey. Also, we made comparisons on accuracy using two different activation functions. The results showed that the ReLU function produced a better performance than the Tanh function most of the time. Though our model is not the best when compared to other models that are already existing, it is simple to understand and interpret and the memory utilized by this model is comparitvely less Also, our model can run smoothly on any normal desktop.

REFERENCES

1. Bullock, Joseph, Carolina Cuesta-Lázaro, and ArnauQuera-Bofarull. "XNet: a convolutional neural network (CNN) implementation for medical x-ray image segmentation suitable for small datasets." Medical Imaging 2019: Biomedical Applications in Molecular, Structural, and Functional Imaging. Vol. 10953. International Society for Optics and Photonics, 2019.

2. T Guo , J Dong, H Li, Y Gao. "Simple convolutional neural network on image classification." 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)(. IEEE, 2017.
3. Thakkar, Vignesh, SumanTewary, and ChandanChakraborty. "Batch Normalization in Convolutional Neural Networks—A comparative study with CIFAR-10 data." 2018 Fifth International Conference on Emerging Applications of Information Technology (EAIT). IEEE, 2018.
4. Sharma, Neha, Vibhor Jain, and Anju Mishra. "An analysis of convolutional neural networks for image classification." Procedia computer science 132 (2018): 377-384.
5. Çalik, Rasim Caner, and M. FatihDemirci. "Cifar-10 Image Classification with Convolutional Neural Networks for Embedded Systems." 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA). IEEE, 2018
6. Krizhevsky, Alex, and Geoff Hinton. "Convolutional deep belief networks on cifar-10." Unpublished manuscript 40.7 (2010): 1-9.
7. Doon, Raveen, Tarun Kumar Rawat, and ShwetaGautam. "Cifar-10 Classification using Deep Convolutional Neural Network." 2018 IEEE Punecon.IEEE.
8. Nazir, Sajid, Shushma Patel, and Dilip Patel. "Hyper Parameters Selection for Image Classification in Convolutional Neural Networks." 2018 IEEE 17th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC). IEEE, 2018.
9. T Liu,S Fang,Y Zaho,P Wang,J Zhang "Implementation of training convolutional neural networks." arXiv preprint arXiv:1506.01195 (2015).
10. Alipourfard, Tayeb, HosseinArefi, and SomayehMahmoudi. "A Novel Deep Learning Framework by Combination of Subspace-Based Feature Extraction and Convolutional Neural Networks for Hyperspectral Images Classification." IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2018.
11. https://medium.com/@kanchansarkar/ReLU-not-a-differentiable-function-why-used-in- gradient-based-optimization-7fef3a4cecec
12. https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6
13. https://github.com/deep-diver/CIFAR10-img-classification-tensorflow

AUTHORS PROFILE

Dr Moulana Mohammed received his Ph.D in Computer Science from Bharathiar University in 2018 and M.Tech in CSE from JNTUK in 2009. He is an Associate Professor in the Computer Science & Engineering, Koneru Lakshmaiah Educational Foundation, India. His research areas include Data mining, Machine Learning, Artificial Intelligence, Big Data Analytics and Deep Learning.

M. Venkata Sai Sowmya pursuing B.Tech in Computer Science & Engineering Department, Koneru Lakshmaiah Educational Foundation, India. Her interests include Deep Neural Networks, Natural Language Processing and Machine Learning. She is an inquisitive person and also a bibliophile.

Y. Akhila pursuing B.Tech in Computer Science & Engineering Department, Koneru Lakshmaiah Educational Foundation, India. Her interests include Deep Learning, Artificial Intelligence, Natural Language Processing and Semantic Analysis. More passionate to do more research.

B. Naga Megana pursuing B.Tech in Computer Science & Engineering Department, Koneru Lakshmaiah Educational Foundation, India. Her interests include Deep Learning and Semantic Analysis.