Binary Image Classification with Deep Convolutional Neural Network using Framework TensorFlow

[1]Kanika Das, [2]Bishwa Ranjan Roy
[1]Student, [2]Faculty
[1,2]Department of Computer Science,
[1,2]Assam University, Silchar, India

Abstract - Image classification is an indispensable tool in the field of computer vision research that can classify an image by evaluating its visual content. To provide an accurate classification of images we need proper algorithms and techniques. Nowadays, Machine Learning has become the most popular subject to provide better techniques for image classification. In this paper, Convolutional Neural Network (CNN) is used to train a binary classification model using Deep learning framework TensorFlow and open source library Keras. A classification model consisting of a large dataset contains 25,000 images of cats and dogs. For training and testing purpose we divided this lager dataset with a ratio of 75:25. Using deep Convoutional Neural Network with activation function Relu. This classification model consists of Convolution filter 3x3, max-pooling filter 2x2. For binary image classification we used Sigmoid activation function with number of epochs 30 and number of steps per epochs 50. After training this Deep Neural Network we achieved a training accuracy of 97.60% and validation accuracy 83.53%.

Index Terms – Machine Learning, Convolutional Neural Network, Image Classification, Deep Learning

I. INTRODUCTION

Artificial intelligence (AI) is an emerging domain of computer science which is capable of incorporating human intelligence in the computer system. The advancements in machine learning and deep learning are creating a paradigm shift in the field of AI research in every sector of the tech industry [1]. Machine Learning [2] is becoming one of the most effective and powerful technologies in for extracting knowledge from the available. There are different approaches of Machine Learning:

1. ***Supervised Machine Learning:*** In this approach, the learning algorithm feed the input in the form of example-label pair to predict the label for each example and takes the feedback whether the prediction is correct or not. [3] In this process of learning, the system improves the accuracy.
2. ***Unsupervised Learning:*** The unsupervised learning predicts without prior knowledge of expected output. The system runs the input data according to the learning algorithm and divides into the multiple groups of similar input to get the insights of input data. The most popular unsupervised learning algorithms are Clustering algorithms and learning algorithms.
3. ***Semi-supervised Learning***: It takes advantages of both Supervised and Unsupervised Learning to produce the desired results. In real world, most of the data are available as a combination of labeled and unlabeled data.
4. ***Reinforced Learning:*** In this learning technique, the system is exposed to an unknown environment to get trained by trial and error method to develop decision making capabilities. The system feed the past experience to make more accurate prediction on the current input.

Convolutional Neural Network is a deep learning approach which is most frequently utilized for processing images and classification applications. It can quickly analyze and spot patterns in images thanks to the layers of convolutional plus pooling procedures that make up its construction. CNNs are highly suited for tasks like image identification and segmentation because they can learn from big datasets. Backpropagation as well as gradient descent are often used to train CNNs, along with optimization methods like Adam and Adadelta. CNNs have achieved success in a variety of diverse fields, such as the field of computer vision, NLP tasks, and categorization of images [2]. Neural networks can learn an approximation of any function f() such that, $y = f(x)$. The non-linear function f is known as the activation function which provides non-linearity behavior into the output of a neuron. In real world, most of the data are non linear in nature so it is important to learn these non linear representations by the neurons. There are several activation functions in practice:

Sigmoid : Sigmoid function predict the probability as an output where the probability of anything lies between 0 and 1. Sigmoid function is represented by the logistic function as follows.

$S(x) = 1 / (1 + \exp(-x))$

Softmax function: Softmax function is used as the activation function in the output layer of the Multi Layer Perceptron which is represented as

Probability (Pass) + Probability (Fail) = 1

tanh : It accepts any real-valued input data and confines it within a range [-1, 1].

$f(x) = \tanh(x) = 2/(1+e^{-2x}) - 1$

ReLU : Rectified Linear Unit (ReLU) accepts a real-valued input with a threshold value zero. It takes negative values and threshold it to zero.

$f(x) = \max(0, x)$

The organization of this paper highlights the purpose of the work in a brief way. In section1, all the basic topics are being covered which includes Artificial Intelligence, Machine Learning, Neural Network. The detailed literature survey is discussed in section 2. In section 3, we have discussed the proposed Classifier Model using CNN algorithm to classify the Image. Result has been discussed in section 4 after getting the desired accuracy. In section 5, Conclusion and Future work is described.

II. LITERATURE SURVEY

Deep learning approaches strive to learn the feature hierarchies created by the combination of lower-level characteristics at higher levels of the hierarchy. In Later 1950s [4], an American psychologist, Frank Rosenblatt developed perceptron that would learn the weights automatically. Initially, perceptron was used in designing electrical machine later the concept is incorporated in computer programming. In 1962, Stuart Dryfus[1] shows a back propagation model to solve numerical problem by using derivative chain rule, instead of dynamic programming.. In 1965, Alexev Grigoryevich Ivakhnenko and Valentin Grigor'evich Lapa uses polynomial activation function to train the machine using Group Method of Data Handling (GMDH) to create a hierarchical representation of Neural Network [3]. This is an introduction to multilayer perception that strengthens the future of deep learning. In 1971, Alexev Grigoryevich Ivakhnenko contributed in the research of neural network by developing 8-layer deep neural network using GMDH [5]. In 1981, Kunihiko Fukushmia proposed Neocognitron [2] which is recognized as the first convolutional neural network architecture capable of recognizing visual pattern such as handwritten character. In 1985, Geoffrey E.Hinton and his team create a Boltzmann machine [6] that is a stochastic recurrent neural network. In 1986, Geoffery E. Hinton implemented back propagation [7], it opens the gate to implement complex Deep neural network. In 1989, Yunn Lecun uses back propagation to train convolutional neural network to recognize handwritten digits [8]. The revolution of Deep Learning continues from developing new algorithms to new hardware. Recently developed algorithm models Generative models [9] to deep minds Alpha go [5]. In this paper, we have design an effective classifier using deep learning framework Tensorflow to achieve better accuracy.

III. METHODOLOGY

In this chapter, we have shown the various stages of our proposed solution in step by step manner using flow graph and each of the state has been discussed in a detailed way. Now, let us study the flow graph as shown in the figure 1.

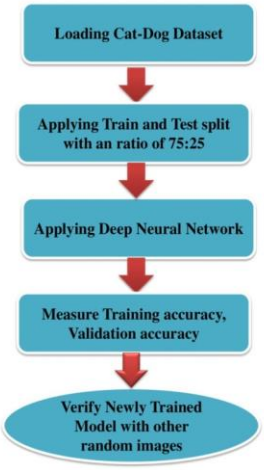

Fig.1 Flow graph of proposed methodology

Input Image Dataset

Import necessary Cat-Dog image dataset. This dataset [10] is consisting total of 25,000 images of Cat and Dog leveled sets. Where, 18750 images of Training set and 6250 images of testing set. Out of this 18750 training images, 9374 images of cat, dog respectively and from 6250 images of testing set 3125 images of cat, dog respectively.

Applying Deep Convulational Neural Network

The steps that go into this process of CNN are broken down as follows

- Step 1 Convolution
- Step 2 ReLU Layer
- Step 3 Pooling
- Step 4 Flattening
- Step 5 Full Connection

Step1 Convolution

In first layer of Convulational Neural Net, the size of convolution matrix is 3x3, with and input shape of an image is 150x150 similarly 2nd, 3rd and 4th layers.

Step 2 ReLU Layer

In our Deep CNN, we choose activation function ReLU, which has a range of zero to +INF. The advantage of applying ReLU is that it gives us 0 values for all the negative values of a matrix.

Step 3 Pooling

We used Max Pooling in our CNN with a matrix of 2x2.

Step 4 Flattening

After max- polling flattening the matrix values into 1-D array.

Step 5 Full Connection

From fully connected deep neural network, we calculate the loss function, cost function and draw a graphical representation of it. Using activation function called sigmoid function we can classify an image is either dog or a cat. Here the following table of parameters used in Deep CNN.

Table of Parameters

Serial Number	Layer (type)	Output Shape	Number of Parameter
1	conv2d (Conv2D)	(None, 148, 148, 16)	448
2	max_pooling2d (MaxPooling2D)	(None, 74, 74, 16)	0
3	conv2d_1 (Conv2D)	(None, 72, 72, 32)	4640
4	max_pooling2d_1 (MaxPooling2)	(None, 36, 36, 32)	0
5	conv2d_2 (Conv2D)	(None, 34, 34, 64)	18496
6	max_pooling2d_2 (MaxPooling2)	(None, 17, 17, 64)	0
7	conv2d_3 (Conv2D)	(None, 15, 15, 128)	73856
8	max_pooling2d_3 (MaxPooling2	(None, 7, 7, 128)	0
9	flatten (Flatten)	(None, 6272)	0
10	dense (Dense)	(None, 512)	3211776
11	dense_1	(Dense) (None, 1)	513

From the above table, we can conclude that total Trainable parameter is 3,309,729.

IV. RESULT AND DISCUSSION

A well known Deep learning algorithm have been use to classify the images whether it's a Cat or a Dog. The dataset is divided into training and testing set into train a deep Convolutional Neural Network with number of epochs 30 and steps per epoch is 50. Using optimizer function RMS prop with learning rate of 0.001 and loss function binary cross entropy. Here, Figure 3 and Figure 4 are the graph of Validation accuracy vs. Training accuracy and Validation loss vs. Training loss are plotted respectively with a training accuracy of 97.60% and validation accuracy 83.53%.

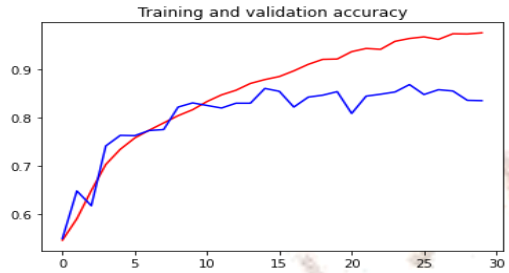

Fig.3 Validation accuracy vs. Training accuracy

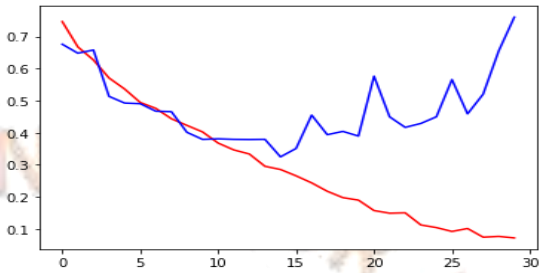

Fig.4 Validation loss vs. Training loss

Here in Figure 3 shows an accuracy of 97.60% as I increased more number of epochs thus Training accuracy also increased a lot. Whereas in Figure 4, it showed overfitting problem. Overfitting happen when training accuracy is high but validation accuracy is less. Which means the model is perfect in training time but when they see some new images for validation purpose then it will fail to predict or less accuracy will occur to predict images.

V. CONCLUSION

We have used Convolutional Neural Networks (CNN) for image classification purpose using images from Kaggle data sets [10]. This gives an image classifier model and with the help of this model image classification can be done efficiently. The data sets used both in training and testing purpose using CNN. The model thus implemented can be extended to a mobile device or any website as per the developer's need for further upgradation. The computational time for processing these images is very high as compared to other normal JPEG image. The model with more layers and training the network with more images will provide more accurate results of classification of images.With an training accuracy of 97.60% and validation accuracy 83.53%. Overfitting is very common problem in Machine learning models. We can solve this overfitting problem by using various methods, one of these methods is called "Image Augmentation". It changed the orientation of images for better understanding of pixels in an image.

VI. REFERENCES

[1] Dreyfus, Stuart. "The numerical solution of variational problems." Journal of Mathematical Analysis and Applications 5.1 (1962): 30-45.

[2] Fukushima, Kunihiko. "Self-organization of a neural network which gives position-invariant response." Proceedings of the 6th international joint conference on Artificial intelligence-Volume 1. 1979.

[3] Farlow, Stanley J. Self-organizing methods in modeling: GMDH type algorithms. Vol. 54. CrC Press, 1984.

[4] Rosenblatt, Frank. "The perceptron: a probabilistic model for information storage and organization in the brain." Psychological review 65.6 (1958): 386.

[5] Holcomb, Sean D., et al. "Overview on deepmind and its alphago zero ai." Proceedings of the 2018 international conference on big data and education. 2018.

[6] Ackley, David H., Geoffrey E. Hinton, and Terrence J. Sejnowski. "A learning algorithm for Boltzmann machines." Cognitive science 9.1 (1985): 147-169.

[7] Plaut, David C., and Geoffrey E. Hinton. "Learning sets of filters using backpropagation." Computer Speech & Language 2.1 (1987): 35-61.

[8] LeCun, Yann, et al. "Backpropagation applied to handwritten zip code recognition." Neural computation 1.4 (1989): 541-551.

[9] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In Advances in neural information processing systems (pp. 2672-2680).

[10] https://www.kaggle.com/datasets