International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 03 | Mar 2019 www.irjet.net p-ISSN: 2395-0072

MACHINE LEARNING BASED OBJECT IDENTIFICATION SYSTEM USING PYTHON

K. Rajendra Prasad[1], P. Chandana Sravani[3], P.S.N. Mounika[3], N. Navya[4], M. Shyamala[5]

[1,2,3,4,5]*Department of Electronics and Communication Engineering, Vignan's Institute of Engineering for Women, Visakhapatnam-46, India*

Abstract- *Nowadays, large amount of data is available everywhere. Therefore, it is very important to analyse this data in order to extract some useful information and to develop an algorithm based on its analysis. This can be achieved through Machine Learning (ML). ML is a subset of Artificial Intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed [9]. As the Human visual system is fast and accurate and can perform complex tasks like identifying multiple objects and detect obstacles with little conscious thought. Our idea is to implement an Object Identification system by using classification type algorithms of ML i.e. Convolution Neural Networks (CNN) in Python. In this project we use different predefined training and test data sets which are used to predict various objects. The main motive of this idea is to identify and obtain the required data of an object by taking the image as input. This is a fundamental approach for many prediction type applications like Self-Driving system.*

***Key Words*: CNN, Machine Learning, Object Identification System, Python, Training Set, Test Set.**

1. INTRODUCTION

Machine learning has been gaining momentum over last decades: self-driving cars, efficient web search, speech and image recognition. The successful results gradually propagate into our daily lives. **ML** is a class of AI methods, which allows the computer to operate in a self-learning mode, without being explicitly programmed.It has wide range of applications in various fields such as Bio-informatics, Intrusion detection, Information retrieval, Game playing, Marketing, Malware detection and Image deconvolution. It is a very interesting and complex topic, which could drive the future of technology [9].

Neural network is a machine learning algorithm, which is built on the principle of the organization and functioning of biological neural networks. It consists of individual units called Neuron. Neurons are located in a series of groups-layers. Neurons in each layer are connected to neurons of the next layer. Data comes from the input layer to the output layer along these compounds. Each individual node performs a simple mathematical calculation. Then it transmits its data to all the nodes it is connected to.

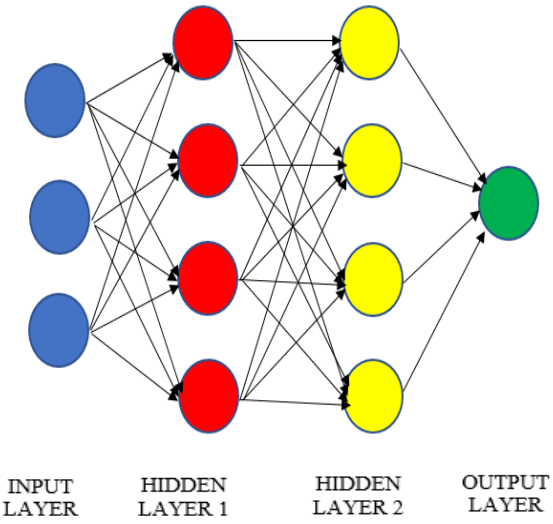

Fig-1: Neural Network

The last wave of neural networks came in connection with the increase in computing power and the accumulation of experience. That brought **Deep learning**, where technological structures of neural networks have become more complex and able to solve a wide range of tasks that could not be effectively solved before [3].

Deep learning is a type of machine learning that requires computer systems to iteratively perform calculations to determine patterns by itself [8]. It aims at learning feature hierarchies with features from higher levels of the hierarchy formed by the composition of lower level features. Automatically learning features at multiple levels of abstraction allow a system to learn complex functions mapping the input to the output directly from data, without depending completely on human-crafted features [1].

One of the major problem in image classification using deep learning is low performance caused due to over fitting. Over fitting happens when the model fits too well to the training set. It then becomes difficult for the model to generalize the new examples that were not in the training set. In order to avoid this problem and to improve performance, large dataset is needed which can be provided by using CNN [2] [6].

Convolutional neural network (CNN) is the frequently used deep learning method for image classification [6]. CNN learns directly from the input image that eliminates manual feature extraction [4].

In this paper, a deep learning convolutional neural network based on keras and tensorflow is developed using python for image classification. Here, we use collection of different images i.e., dataset which contains two types of animals, namely cat and dog which are used to train the system.

Fig-2: Sample Dataset

In this paper, different classifiers such as softmax, sigmoid in combination with an activation function Relu of CNN are compared.

Tensorflow is an open-source software library for dataflow programming across a range of tasks. It is used for image classification in deep learning and also for machine learning applications such as neural networks.

Keras is an open source neural network library written in python. It is designed to enable fast experimentation with deep neural networks.

2. BASIC THEORY

2.1. Convolution Neural Network

In CNN, the neuron in a layer is only connected to a small region of the layer before it, instead of all the neurons in a fully connected manner, so CNN handle fewer amounts of weights and also less number of neurons. In machine learning, convolution neural networks are complex feed forward neural networks. CNNs are used for image classification and recognition because of its high accuracy [4].

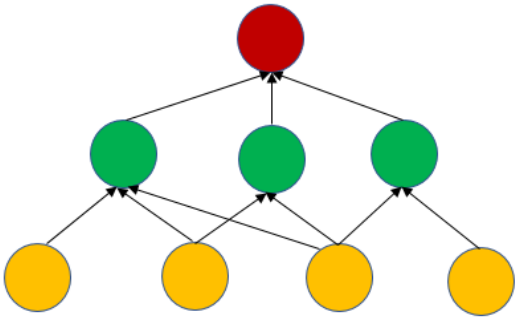

Fig-3: Convolutional Neural Network

2.2. Classifier Functions:

Classifiers are used when we would want our neural networks to work on complicated tasks like language translations and image classifications.

2.2.1. Sigmoid or Logistic Classifier:

The Sigmoid Function curve looks like S-shape.

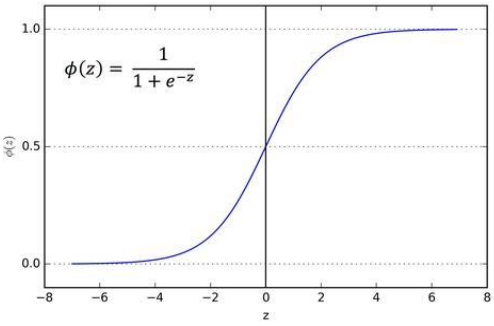

Fig-4: Sigmoid Function

The main reason why we use sigmoid function is because it exists between (**0 to 1).**Therefore, it is especially used for models where we have to **predict the probability** as an

output. Since probability of anything exists only between the range of **0 and 1,** sigmoid is the right choice.

2.2.2. Softmax Classifier:

The softmax function is also a type of sigmoid function but is handy when we are trying to handle classification problems. The sigmoid function as we saw earlier was able to handle just two classes. What shall we do when we are trying to handle multiple classes? Just classifying yes or no for a single class would not help then. The softmax function would squeeze the outputs for each class between 0 and 1 and would also divide by the sum of the outputs. This essentially gives the probability of the input being in a particular class. It can be defined as

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \text{for j = 1, 2, 3 ……., K.} \quad \text{- eq(1)}$$

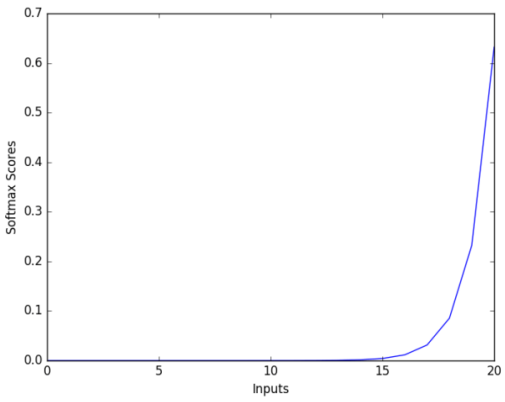

Fig-5: Softmax Function

Let's say for example we have the outputs as-

[1.2, 0.9, 0.75], When we apply the softmax function we would get [0.42, 0.31, 0.27]. So now we can use these as probabilities for the value to be in each class.

The softmax function is ideally used in the output layer of the classifier where we are actually trying to attain the probabilities to define the class of each input.

2.3. Activation Function

They basically decide whether a neuron should be activated or not. Whether the information that the neuron is receiving is relevant for the given information or should it be ignored.

$$Y = \text{Activation}(\Sigma(weight * input) + bias) \quad \text{- eq(2)}$$

The activation function is the non-linear transformation that we do over the input signal. This transformed output is then sent to the next layer of neurons as input.

When we do not have the activation function the weights and bias would simply do a linear transformation. A linear equation is simple to solve but is limited in its capacity to solve complex problems. A neural network without an activation function is essentially just a linear regression model. The activation function does the non-linear transformation to the input making it capable to learn and perform more complex tasks. We would want our neural networks to work on complicated tasks like language translations and image classifications. Linear transformations would never be able to perform such tasks.

2.3.1. ReLU (Rectified Linear Unit) Activation Function

The ReLU is the most used activation function in the world right now. Since, it is used in almost all the convolutional neural networks or deep learning.

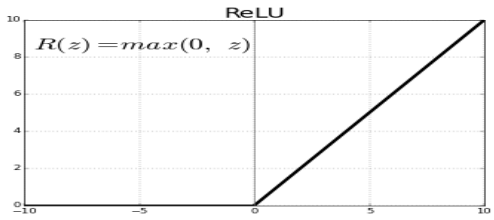

Fig-6: ReLU Function

As you can see, the ReLU is half rectified (from bottom). f(z) is zero when z is less than zero and f(z) is equal to z when z is above or equal to zero.

Range: [0 to infinity)

3. EXPERIMENTAL SETUP

In this paper, we perform experiments on windows 10 in python 3.7 on CPU system and create the CNN model based on keras and tensorflow libraries. The CNN model used for experiments is shown in fig 7. This model mainly consists of four layers including, convolutional, pooling, flattening and fully connected layers.

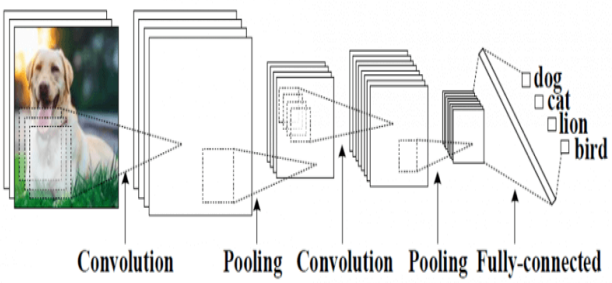

Fig-7: Structure of CNN

The convolution function is taking 4 arguments, the first is the number of filters i.e., 32, the second is the shape of each filter of size 3x3, the third is the shape of type of image (RGB or black and white) of each image i.e., the input to CNN is 64x64 resolution and 3 stands for RGB, the fourth argument is the activation function.

We need to perform pooling operation on the resultant feature maps in order to compress the size of image using a window of size 2x2 pixels.

For performance measurement we use activation function namely, ReLU (Rectified linear unit), and two classifiers namely Softmax, Sigmoid. In experiment, we use combination of activation function with different classifiers in building the identification system, and analyze that which combination gives better accuracy for image identification.

After implementing all above parameters in python, we train and test CNN model using training and test datasets, and then obtain accuracy for different CNN structures. After then we compare the obtained accuracies and finds which CNN structure results in higher accuracy.

4. RESULTS

The obtained accuracies of different CNN structures are listed in the below Table-1 [5].

Table-1: Showing the Results

Number of convolutional layers	Activation Function	Classifier	Classification Accuracy
2	ReLU	Sigmoid	90.56
2	ReLU	Softmax	50.96

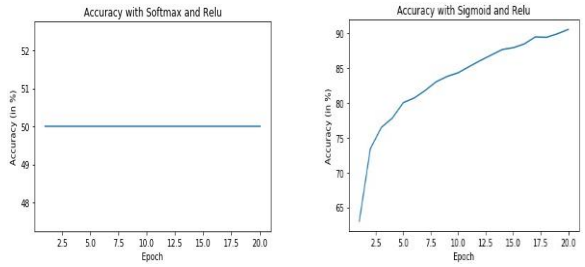

Fig-8: Graphical Representation of accuracy vs number of epochs for different Classifier functions.

5. CONCLUSIONS

In this paper, a deep learning convolutional neural network based on keras and tensorflow is developed using python 3.7 for image classification. Here, we compared two different structures of CNN, with different combinations of classifier and activation function.

From experiments, we obtained results for each combination and observed that ReLU activation function and sigmoid classifier combination gives better classification accuracy (nearly 90.5% [5]) than any other combination of activation function and classifier.

ACKNOWLEDGEMENTS

The Authors express the deepest gratitude to Dr.J.Sudhakar, Head of the Department, Department of Electronics and Communication Engineering, for his cooperation and encouragement during the present work. We express our sincere thanks to AUTHOURITIES of Vignan's Institute of Engineering for Women, for their cooperation in extending the necessary facilities to carry out our research work.

REFERENCES

[1] Sameer Khan and Suet-Peng Yong "A Deep Learning Architecture for Classifying Medical Image of Anatomy Object", Annual Summit and Conference, ISBN 978-1-5386-1543-0, pp. 1661-1668, 2017

[2] Rui Wang, Wei Li, Runnan Qin and JinZhong Wu "Blur Image Classification based on Deep Learning", IEEE, ISBN 978-1-5386-1621-5 pp. 1-6, 2017

[3] TenyHandhayani, JansonHendryli, Lely Hiryantyo "Comparison of Shallow and Deep Learning Models for Classification of Lasem Batik Patterns", ICICoS, ISBN 978-1-5386-0904-0, pp. 11-16, 2017

[4] Laila Ma‟rifatulAzizah, SittiFadillahUmayah, SlametRiyadi, CahyaDamarjati, NafiAnandaUtama "Deep Learning Implementation using Convolutional Neural Network in Mangosteen Surface Defect Detection", ICCSCE, ISBN 978-1-5386-3898-9, pp. 242-246, 2017

[5] Sebastian Stabinger, Antonio Rodr´ıguez-S´anchez "Evaluation of Deep Learning on an Abstract Image Classification Dataset", IEEE International Conference on Computer Vision Workshops (ICCVW), ISBN 978-1-5386-1035-0, pp. 2767-2772, 2017

[6] Hasbi Ash Shiddieqy, FarkhadIhsanHariadi, Trio Adiono "Implementation of Deep-Learning based Image Classification on Single Board Computer", International Symposium on Electronics and Smart Devices (ISESD), ISBN 978-1-5386-2779-2, pp. 133-137, 2017

[7] Hangning Zhou, FengyingXie, Zhiguo Jiang, Jie Liu, Shiqi Wang, Chenyu Zhu "Multi-classification of skin diseases for dermoscopy images using deep learning", International Conference on Imaging Systems and Techniques (IST), ISBN 978-1-5386-1621-5, pp. 1-5, 2017

[8] Sachchidanand Singh, Nirmala Singh "Object Classification to Analyze Medical Imaging Data using Deep Learning", International Conference on Innovations in information Embedded and Communication Systems (ICIIECS), ISBN 978-1-5090-3295-2, pp. 1-4, 2017

[9] Kodratoff, Y., Introduction to Machine Learning, Pitman, London 1988.

[10] Kodratoff, Y., R. Michalski (Eds.): Machine Learning An Artificial Intelligence Approach Vol. III, Morgan Kaufmann, San Mateo, 1990.